

17.658/H/03



APLIKASI KONTROLER ADAPTIF NEURO FUZZY PADA PENGATURAN KECEPATAN MOTOR DC

Oleh:

A. EJAH UMRAENI SALAM

NRP. 2299 202 006



RT
629.89
Sal
a-1
2002

BIDANG KEAHLIAN TEKNIK SISTEM PENGATURAN
PROGRAM STUDI TEKNIK ELEKTRO
PROGRAM PASCASARJANA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA, 2002

PERPUSTAKAAN ITS	
Tgl. Terima	21 - 8.2002
Terima Dari	H
No. Agenda Prp.	25950

...Takutlah engkau kepada Allah dimana saja kamu berada, dan
ikutilah sesuatu kejahatan itu dengan kebaikan, maka Ia akan
menghapusnya, dan bergaulah dengan sesama manusia dengan
akhlak yang baik.
(HR. Tirmidzi)

Lembar Pengesahan

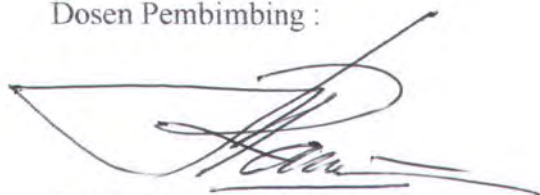
APLIKASI KONTROLER ADAPTIF NEURO FUZZY PADA PENGATURAN KECEPATAN MOTOR DC

**Tesis ini disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Teknik (M.T.)
di
Institut Teknologi Sepuluh Nopember**

Oleh :

**A. Ejah Umraeni Salam
Nrp. 2299 202 006**

Dosen Pembimbing :



1. **DR.IR. MOCH. RAMELI**
NIP. 130 938 507

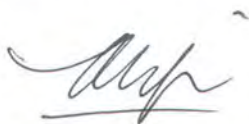
Disetujui oleh Tim Penguji Tesis :

Tanggal Ujian : 14 Agustus 2002

Periode Wisuda : September 2002



1. **IR. RUSDHANTO EFFENDI AK, MT**
NIP. 131 453 676



2. **IR. ALI FATHONI, MT**
NIP. 131 842 499

Direktur Program Pascasarjana :



DR.IR. SEKARTEDJO, M.Sc.
NIP. 130 701 281

...Dan ucapkanlah kepada ibu-bapakmu perkataan yang mulia dan rendahkanlah dirimu terhadap keduanya dengan penuh kasih sayang dan do'akanlah :

“Wahai Tuhanku, kasihanilah mereka keduanya, sebagaimana mereka berdua telah mendidik aku di waktu kecil”.

(Al-Israa' : 23-24)

Abstrak

APLIKASI KONTROLER ADAPTIF NEURO FUZZY PADA PENGATURAN KECEPATAN MOTOR DC

Oleh : A. Ejah Umraeni
Dosen Pembimbing : Dr.Ir.Moch.Rameli

Pada penelitian ini dikembangkan suatu metode pengendalian yang memadukan antara sistem adaptif dengan jaringan syaraf tiruan serta kontroler logika fuzzy sehingga disebut kontroler ANF (Adaptive-Neuro-Fuzzy) untuk mengatur dc-dc konverter dalam pengaturan kecepatan motor dc.

Sistem pengendalian yang digunakan adalah sistem pengaturan berbasis komputer, sehingga pada saat terjadi gangguan dapat diperoleh respon pengaturan kecepatan dengan performansi yang baik.

Respon yang baik diperoleh apabila kontroler dapat menyesuaikan dengan kondisi yang sedang terjadi, hal ini ditunjukkan dengan kemampuan kontroler ANF untuk menentukan sendiri atau membangkitkan secara otomatis fungsi keanggotaannya (membership function) dengan terlebih dahulu melalui proses pembelajaran. Sistem logika fuzzy menggunakan fungsi koridor untuk menentukan fungsi keanggotaannya. Pada penelitian ini terbatas pada skala simulasi yang dinyatakan dalam gambar karakteristik respon yang terjadi untuk perubahan setpoint dan perubahan beban.

Kata Kunci : Adaptive neuro fuzzy (ANF), fungsi koridor, identifikasi parameter, jaringan syaraf tiruan, motor dc penguatan terpisah, dc-dc konverter,

APLICATION CONTROLLER ADAPTIF NEURO-FUZZY TO CONTROL THE SPEED OF THE DC MOTOR

By : A. Ejah Umraeni Salam
Under the supervision : Dr.Ir. Moch. Rameli

This research is expanded a methode controlled are integrated between adaptive system, neural network and fuzzy logic controlled. The named is ANF controlled is used at dc Converters in speed control motor dc.

The control system is used in control system based computer, and if disturbance exist can be obtained respons control system with good performance.

Good respons is obtained if control can be adapt with this all condition. That is indicated if ANF controlled capability to provoked in automatic membership function by means of learning process. The fuzzy logic controlled is used corridor function to established membership function. This research limited scala simulation and describred in graphic respons characteristic to setpoint changed and load changed.

Key word : Adaptive neuro-fuzzy (ANF), corridor function, identification parameter, artificial neural-network, dc motor excited separate, dc converters.

“Dan kami perintahkan kepada manusia agar berbuat baik kepada ibu-
bapaknya. Ibunya mengandungnya dalam keadaan lemah yang
bertambah-tambah...”
(Luqman :14)

Kata Pengantar

KATA PENGANTAR

Alhamdulillahirobbil 'aalamiin, tiada kata yang lebih pantas diucapkan seorang hamba selain puji dan syukur ke hadirat-Nya, yang senantiasa menolong dan memberi petunjuk pada hamba-Nya. Karena dengan pertolongan Allah SWT tesis ini akhirnya bisa terselesaikan.

Tesis ini disusun untuk memenuhi salah satu persyaratan dalam menyelesaikan pendidikan strata-2 pada Bidang Keahlian Teknik Sistem Pengaturan Program Studi Teknik Elektro Institut Teknologi Sepuluh Nopember Surabaya.

Penulis mohon maaf atas kekurangan dalam tesis ini, dan berharap dapat bermanfaat dalam pengembangan selanjutnya. Kritik dan saran dari semua pihak sangat penulis harapkan demi kesempurnaan tesis ini.

Pada kesempatan ini, penulis menyampaikan ucapan terima kasih dan penghargaan yang sebesar-besarnya kepada :

1. **Bapak DR. Ir. Moch. Rameli.**, selaku dosen pembimbing yang dengan penuh ketelatenan telah memberikan pengarahan dan bimbingannya.
2. **Bapak Ir. Rusdhianto Effendi, AK, MT.**, yang dengan keikhlasan dan kesabarannya memberikan petunjuk-petunjuk yang sangat berharga dalam menyelesaikan thesis ini.
3. **DR. Ir. Achmad Jazidie, M.Eng.**, selaku Ketua Program Studi Teknik Elektro.

4. **Seluruh Dosen Teknik Sistem Pengaturan Teknik Elektro** yang telah mengajar, mendidik dan membimbing kami
5. **Staf Administrasi Teknik Elektro** yang banyak membantu kami dalam urusan administrasi
6. **Rekan-rekan sewaktu kuliah di Teknik Sistem Pengaturan (Aziz, Hadi, Pri, Puput)** yang banyak sekali membantu dan mendorong kami.
7. **Suami tercinta dan anak-anakku** tersayang atas segala bantuan dan dorongannya lahir dan bathin.
8. **Kedua orang tua , saudara-saudara**, atas segala dorongan moril, pengertian serta yang terutama bantuan do'anya
9. Semua pihak yang penulis tidak sempat sebutkan satu per satu yang telah membantu kami, baik secara langsung maupun tidak langsung.

Semoga Allah SWT akan melimpahkan berkat dan rahmat-Nya atas semua bantuannya. Amiin.

Surabaya, Agustus 2002

Penulis

... Shodaqoh tidak menyebabkan berkurangnya harta, memberikan maaf kepada orang yang bersalah akan menambah kemuliaan. Barangsiapa merendahkan dirinya untuk mencari ridho Allah, Allah akan memuliakannya ...
(HR. Muslim)

Daftar Isi

DAFTAR ISI

Halaman

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Batasan Masalah	3
1.4. Tujuan Penelitian	3
1.5. Kontribusi Penelitian	3
BAB II TINJAUAN PUSTAKA	4
2.1. Struktur Model Sistem	4
2.2. Identifikasi Parameter Sistem Adaptif	6
2.3 Jaringan Syaraf Tiruan	9
2.3.1. Pemodelan Jaringan Syaraf Tiruan	10
2.3.2. Arsitektur Jaringan Syaraf Tiruan	12
2.3.3 Algoritma Belajar	13
2.3.4 Algoritma Back Propagation	14
2.3.5 Tahapan pada Algoritma Back Propagation	15
2.3.5.1 Propagasi Maju	16
2.3.5.2 Propagasi Balik	17
2.3.6. Jaringan Pada Lapisan Output	18
2.3.7 Jaringan Pada Lapisan Hidden	19

2.4.	Kontroler Logika Fuzzy	21
2.4.1.	Struktur Dasar KLF	21
2.4.2.	Fuzzyfikasi	22
2.4.3.	Basis Data	23
2.4.4.	Aturan Dasar	25
2.4.5.	Logika Pengambilan Keputusan	29
2.4.6.	Defuzzifikasi	30
2.4.7.	Perubahan Rnge Fungsi Keanggotaan dengan Fungsi Koridor	32
2.5	DC-DC Konverter	34
2.5.1.	Prinsip operasi dan teknik kontrol dc-dc konverter	35
2.6.	Motor DC Penguat Terpisah	36
2.7.	Spesifikasi Respon	41
BAB III	METODE PENELITIAN	44
3.1.	Langkah-langkah Penelitian	44
3.1.1.	Pemodelan Sistem	44
3.1.2.	Identifikasi Parameter SistemPlant	44
3.1.3.	Mencari nilai alfa untuk setiap perubahan beban	46
3.1.4.	Pembelajaran Jaringan Syaraf Tiruan	46
3.1.5.	Simulasi Sistem Keseluruhan	47
3.2.	Metode Pengujian	48
BAB IV	PENGUJIAN DAN ANALISIS DATA	49
4.1.	Pengujian Dengan Kontroler Fuzzy Biasa	49
4.2.	Pengujian Dengan Kontroler Adaptif Neuro Fuzzy	53
4.2.1.	Proses Identifikasi dan Pembelajaran	53
4.2.2.	Pengujian dengan Adaptif Neuro-Fuzzy	55
4.3.	Pengujian Dengan Perubahan Setpoint	58
4.4.	Pengujian Dengan Perubahan Beban	60
BAB V	KESIMPULAN DAN SARAN-SARAN	65
5.1.	Kesimpulan	65
5.2.	Saran-saran	65

DAFTAR PUSTAKA	66
LAMPIRAN A	67
LAMPIRAN B	68
LAMPIRAN C	69

“ Ya Allah Tuhan yang maha Hidup, yang Berdiri sendiri, Pencipta langit dan bumi,
Tuhan yang maha Besar dan maha Mulia ya Allah
tidak ada Tuhan yang patut disembah hanya Engkau, aku memohon kepadamu
semoga kiranya Engkau hidupakan akan hatiku dengan nur hidayah-Mu.
Ya Tuhanku yang maha Kasih Sayang.
(Risalah Do'a)

Daftar Gambar

DAFTAR GAMBAR

<i>No. Gambar</i>	<i>Nama Gambar</i>	<i>Hal.</i>
2-1.	Model sel syaraf	10
2-2.	Kurva output fungsi sigmoid	11
2-3.	Arsitektur jaringan syaraf tiruan	13
2-4.	Struktur Dasar KLF	22
2-5.	Tanggapan Sistem Loop Tertutup	27
2-6.	Max-Min fuzzy reasoning	30
2-7.	Interpretasi Grafik dari Strategi Defuzzifikasi	31
2-8.	Perubahan range fungsi keanggotaan dengan fungsi koridor	32
2-9.	Fungsi keanggotaan uniform ($\alpha = 1$)	33
2-10.	Fungsi Keanggotaan yang telah dimodifikasi ($\alpha = 2$)	34
2-11.	Dasar rangkaian dc-dc konverter step down (klas A)	35
2-12.	Model motor DC dengan penguatan terpisah	37
2-13.	Rangkaian jangkar motor dc	39
2-14.	Hubungan torsi dengan beban	39
2-15.	Blok Diagram motor dc penguatan terpisah dengan pengaturan jangkar	41
2-16.	Spesifikasi Respon Transien	43
3-1.	Blok Diagram untuk mencari nilai alfa	46
3-2.	Arsitektur jaringan syaraf tiruan untuk pembelajaran nilai alfa	46
3-3.	Blok diagram system pengaturan kecepatan motor dc dengan kontroler ANF	47
4-1.	Blok Diagram system pengaturan kecepatan motor DC dengan kontroler Fuzzy Biasa	50
4-2.	Karakteristik respon dan aksi kontrol untuk $\alpha = 1$	50

4-3.	Fungsi koridor untuk $\alpha = 1$	51
4-4.	Fungsi keanggotaan untuk $\alpha = 1$	51
4-5.	Karakteristik respon dan aksi kontrol untuk $\alpha = 0,5$	52
4-6.	Fungsi koridor untuk $\alpha = 0,5$	52
4-7.	Fungsi keanggotaan untuk $\alpha = 0,5$	53
4-8.	Blok Diagram untuk identifikasi Parameter	54
4-9.	Blok Diagram untuk mencari nilai α	54
4-10.	Kurva training ANN mencari hubungan α dan beban	55
4-11.	Blok Diagram system Pengaturan kecepatan motor dc dengan kontroler ANF	56
4-12.	Karakteristik respon, aksi kontrol, fungsi koridor dan fungsi keanggotaan setelah pembelajaran	57
4-13.	Karakteristik respon dan aksi kontrol untuk perubahan setpoint	58
4-14.	Karakteristik respon dan aksi kontrol untuk perubahan setpoint setelah pembelajaran	59
4-15	Karakteristik respon dan aksi kontrol untuk perubahan beban yang tetap	60
4-16	Karakteristik respon dan aksi kontrol untuk perubahan beban sesaat	61
4-17	Karakteristik respon dan aksi kontrol untuk perubahan beban yang tetap setelah pembelajaran	62
4-18	Karakteristik respon dan aksi kontrol untuk perubahan beban sesaat setelah pembelajaran	63

... Shodaqoh tidak menyebabkan berkurangnya harta, memberikan maaf kepada orang yang bersalah akan menambah kemuliaan. Barangsiapa merendahkan dirinya untuk mencari ridho Allah, Allah akan memuliakannya ...
(HR. Muslim)

DAFTAR TABEL

DAFTAR TABEL

<i>No. Tabel</i>	<i>Nama Tabel</i>	<i>Hal.</i>
2-1.	Prototipe Aturan Kontrol Linguistik dengan 3 Nilai	28
2-2	Penyempurnaan Aturan Konrol dengan 3 Nilai	28
2-3	Tabel Keputusan Aturan Kontrol Fuzzy Mac Vicar-Whean	29
4-1	Kriteria Performansi untuk beberapa Pengujian	58
4-2	Kriteria performansi untuk pengujian Kontroler untuk perubahan setpoint dan beban	64

"Ya Tuhan kami, janganlah Engkau jadikan hati kami condong kepada kesesatan sesudah Engkau beri petunjuk kepada kami, dan karuniakanlah kepada kami rahmat dari sisi Engkau; karena sesungguhnya Engkau lah Maha Pemberi karunia".
(Ali Imran : 8)

BAB I

PENDAHULUAN

BAB I

PENDAHULUAN

I.1 LATAR BELAKANG

Kebutuhan akan pengaturan daya elektrik dan sistem penggerak motor listrik pada industri telah ada sejak lama dan semakin berkembang sesuai dengan perkembangan teknologi dewasa ini. Dan pada banyak aplikasi industri, diperlukan untuk mengubah sumber tegangan dc tetap menjadi sumber tegangan dc yang bersifat variable. Dc chopper mengubah secara langsung dari dc ke dc dan disebut juga dengan DC-DC konverter. Dc-dc konverter ini digunakan untuk mengontrol kecepatan putaran motor dc untuk keperluan mesin produksi..

Salah satu cara yang digunakan untuk pengaturan dc-dc konverter untuk mengatur kecepatan motor dc yaitu dengan menggunakan Adaptif-Neurol-Fuzzy (ANF) . Yaitu penggabungan antara metoda jaringan syaraf tiruan (neural network) dan kontroler logika fuzzy. Dan selanjutnya kontroler logika fuzzy memiliki kemampuan memodifikasi fungsi keanggotaan melalui fungsi koridor sehingga dapat menghasilkan output sesuai dengan performansi yang diinginkan. Dan pada penulisan ini dilakukan penerapan adaptif neuro-fuzzy pada pengaturan dc-dc konverter untuk mengatur putaran motor dc sehingga nantinya dapat menghasilkan output sesuai dengan performansi yang diinginkan.

1.2. PERUMUSAN MASALAH

Untuk suatu kebutuhan tertentu sering kali diperlukan motor dc dengan kecepatan yang terjaga konstan walaupun dikenakan beban yang berubah-ubah dan kadangkala justru diinginkan kecepatan motor yang berubah-ubah besarnya, tetapi perubahan tersebut sesuai dengan harga-harga yang telah ditetapkan terlebih dahulu. Oleh karena itu salah satu cara yang digunakan yaitu dengan mengatur besarnya tegangan DC pada kumparan medan atau jangkar dengan menggunakan rangkaian DC-DC Konverter . Dan untuk pengaturannya digunakan Adaptif-Neuro-Fuzzy (ANF). Sehingga aksi kontrol yang dihasilkan oleh kontrol adaptif neuro-fuzzy menjadi input bagi DC-DC Konverter untuk menghasilkan tegangan jangkar yang dapat menggerakkan motor dc. Dalam hal ini tegangan medan dianggap konstan.

Untuk itu hal yang perlu dilakukan terlebih dahulu adalah mengidentifikasi parameter dari plant yang akan diatur dan outputnya berupa parameter-parameter yang berubah-ubah akan menjadi masukan bagi jaringan syaraf tiruan. Dan dengan metode back-propagation diperoleh hasil berupa nilai alpha (α) akan menjadi input dari fungsi koridor untuk mengubah fungsi keanggotaan . Dengan menggunakan fungsi koridor ini diharapkan nantinya dapat memperoleh output berupa aksi kontrol untuk input dc-dc konverter dalam mengatur putaran motor dc sesuai dengan performansi yang diinginkan.

Dengan memadukan kedua metode yaitu back propagation neural-network dan kontrol logika fuzzy dalam mendisain kontroler adaptif neuro fuzzy diharapkan dapat diperoleh hasil yang lebih memuaskan disbanding dengan menggunakan fuzzy biasa.

1.3. BATASAN MASALAH

Untuk merealisasikan disain kontroler adaptif-neuro-fuzzy (ANF) perlu pembatasan sebagai berikut :

1. Plant yang digunakan adalah dc-dc konverter step-down dan motor dc penguatan terpisah.
2. Mendisain kontroler adaptif- neuro-fuzzy yang direpresentasikan secara perangkat lunak (software) dalam bentuk simulasi
3. Menggunakan fungsi koridor bentuk segitiga dalam memodifikasi fungsi keanggotaan dan metode Mac Vicar Whelan yang digunakan dalam mendisain rule-rule base pada ANF.

1.4. TUJUAN PENELITIAN

Tujuan dari penulisan ini secara umum adalah untuk mempelajari dan mengetahui sejauhmana kemampuan yang dimiliki oleh kedua teknologi jaringan syaraf tiruan dan kontrol logika fuzzy adaptif ketika dipadukan terutama dalam mengatasi perubahan setpoint dan beban yang mungkin terjadi pada saat digunakan untuk memperoleh kecepatan putaran motor dc sesuai dengan yang performansi yang diinginkan.

1.5. KONTRIBUSI

Dengan kontroler ANF diharapkan dapat digunakan dalam mengatur kecepatan Motor DC dengan hasil yang lebih memuaskan dibanding kontroler fuzzy Biasa

“Ya Allah, tidak ada yang mudah, kecuali sesuatu yang Engkau jadikan mudah, Engkaulah yang menjadikannya yang susah itu mudah”.
(Risalah Do’a)

BAB II

TINJAUAN PUSTAKA

BAB II

TINJAUAN PUTAKA

2.1. STRUKTUR MODEL SISTEM

Jika yang digunakan adalah model SISO, linier dan time invariant. Maka sistem kontinu dapat dinyatakan dalam persamaan differensial sebagai berikut:

$$a_n \frac{d^n}{(dt)^n} y(t) + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t) = b_m \frac{d^m u(t)}{(dt)^m} + \dots + b_1 \frac{du(t)}{dt} + b_0 u(t) \quad (2.1)$$

atau dalam domain kompleks persamaan diatas ditulis sebagai berikut

$$Y(s) = H(s) \cdot U(s) \quad (2.2)$$

Dimana fungsi alih sistem $H(s)$ diberikan

$$H(s) = \frac{B(s)}{A(s)} = \frac{b_0 + b_1 s + \dots + b_m s^m}{a_0 + a_1 s + \dots + a_n s^n} \quad (2.3)$$

Secara diskrit

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z + \dots + b_m z^m}{a_0 + a_1 z + \dots + a_n z^n} \quad (2.4)$$

sehingga :

$$Y(z) = \frac{B(z^{-1})}{A(z^{-1})} U(z) \quad (2.5)$$

Dengan polinom z , $A(z^{-1})$ dan $B(z^{-1})$ didefinisikan sebagai

$$A(z^{-1}) = 1 + \sum_{i=1}^{nA} a_i z^{-i} \quad (2.6)$$

$$B(z^{-1}) = \sum_{i=0}^{nB} b_i z^{-i} \quad (2.7)$$

Dimana nA dan nB masing-masing adalah orde dari polinom z , $A(z^{-1})$ dan $B(z^{-1})$. Dan dalam hal sistem dengan faktor delay d satu satuan waktu periode sampling sehingga persamaan ditulis :

$$Y(z) = \frac{z^d B(z^{-1})}{A(z^{-1})} U(z) \quad (2.8)$$

Jika kemudian polinom z persamaan (2.6) dan (2.7) disubstitusikan kedalam model kompleks, maka diperoleh :

$$y(z) + a_1 z^{-1} y(z) + \dots + a_{nA} z^{-nA} y(z) = b_1 z^{-1-d} u(z) + \dots + b_{nB} z^{-nB-d} u(z) \quad (2.9)$$

Dan transformasi z baliknya diperoleh persamaan diskrit dalam domain waktu :

$$y(k) + a_1 y(k-1) + \dots + a_{nA} y(k-nA) = b_1 u(k-1-d) + \dots + b_{nB} u(k-nB-d) \quad (2.10)$$

Suatu sistem dinamik dapat juga dimodelkan dengan persamaan regresi linier sebagai berikut :

$$y(k) = \theta^T \phi(k-1) + \varepsilon(k) \quad (2.11)$$

dimana :

θ adalah vektor parameter

$\phi(k-1)$ adalah vector informasi yang berisikan data input output terdahulu

$\varepsilon(k)$ adalah gangguan (disturbance)

Untuk sistem pada persamaan diatas, diperoleh harga vektor parameter dan vektor informasi sebagai berikut :

$$\theta = (a_1 \dots a_{nA} \quad b_1 \dots b_{nB})$$

$$\phi(k-1) = [-y(k-1), \dots, -y(k-n_A) \quad u(k-1-d), \dots, u(k-n_B-d)]$$

Vektor parameter ini yang hendak ditentukan dengan identifikasi parameter sistem.

2.2. IDENTIFIKASI PARAMETER SISTEM ADAPTIF

Banyak teknik perancangan sistem yang dapat diterapkan untuk mendisain suatu sistem diantaranya adalah teknik perancangan dengan kontrol adaptif. Parameter kontrol yang berubah mengikuti perubahan (adaptasi) parameter proses merupakan ciri kerja sistem kontrol adaptif, karenanya untuk melakukan perubahan pada parameter kontrol diperlukan informasi tentang parameter proses, atau dengan kata lain perlu dilakukan identifikasi dari parameter sistem.

Informasi yang diperlukan untuk identifikasi parameter sistem adalah sebagai berikut :

(1) Struktur sistem.

Hubungan antar parameter, yaitu keterkaitan antara parameter sistem berikut noise yang ada.

(2) Orde sistem (ditentukan oleh derajat polinomial).

(3) Parameter sistem (letak parameter pada polinomial).

Identifikasi parameter system disini menggunakan prinsip kuadrat terkecil (least square) yaitu metode yang meminimumkan jumlah kuadrat dari keluaran. Identifikasi dilakukan dalam waktu riil yang diharapkan dapat memberikan hasil estimasi secara rekursif dengan hasil pengukuran yang diperoleh. Estimasi parameter digunakan untuk memperoleh parameter-parameter proses yang tidak diketahui atau

yang berubah terhadap waktu. Metode yang digunakan untuk identifikasi parameter sistem dalam penulisan ini adalah Metode Rekursif Least Square.

Diketahui persamaan regresi linier adalah sebagai berikut :

$$y(k) = \theta^T \phi(k-1) + \epsilon(k)$$

dimana :

$$\theta = (a_1, \dots, a_{nA}, b_1, \dots, b_{nB})$$

$$\phi(k-1) = [-y(k-1), \dots, -y(k-nA), u(k-1-d), \dots, u(k-nB-d), e(k-1), \dots, e(k-nC)]$$

d adalah delay antara input dan output

Karena $e(k-1) \dots e(k-nC)$ tidak mudah diukur, maka error yang mungkin diprediksi dengan cara sebagai berikut :

$$\text{Prediksi Error: } \epsilon_e(k) = y(k) - \hat{y}_e(k) = y(k) - \hat{\theta}^T(k-1)\phi(k-1) \quad (2.12)$$

dimana :

$y(k)$ adalah output yang terukur

$\hat{y}_e(k)$ adalah estimasi tanpa noise

$F(k)$ adalah gain adaptasi yang besarnya :

$$F(k+1) = F(k) - \frac{F(k)\phi(k)^T \phi_e(k) F(k)}{1 + \phi_e(k)^T F(k) \phi(k)} \quad (2.13)$$

Dalam implementasi $\phi(k) = \phi_e(k) \Big|_{e(k) = \epsilon(k)}$

Pada estimasi dengan RLS (recursive least square), gain adaptasi berubah semakin kecil. Hal ini sangat penting, karena jika gain adaptasi sudah sedemikian kecilnya sedang perubahan masih terjadi (sistem non stasioner), maka proses estimasi tidak mempunyai kemampuan lagi dalam mengadaptasi perubahan tersebut.

Untuk mengatasi persoalan tersebut dilakukan modifikasi terhadap algoritma pengabai (forgetting factor) α , sehingga penulisan model umum penguatan adaptasi menjadi sebagai berikut :

$$F(k)^{-1} = \alpha_1(k) F(k-1)^{-1} + \alpha_2(k) \phi_e(k-1) \phi_e(k-1)^T \quad (2.14)$$

Dengan : $0 < \alpha_1 \leq 1$; $0 \leq \alpha_2 < 2$; $F(0) > 0$ dengan perumusan diatas, diperoleh algoritma rekursif penguatan adaptasi sebagai berikut :

$$F(k+1) = \frac{1}{\alpha_1(k)} \left\{ F(k) - \frac{F(k) \phi(k)^T \phi_e(k) F(k)}{\beta(k) + \phi_e(k)^T F(k) \phi(k)} \right\} \quad (2.15)$$

$$\text{Dengan } \beta(k) = \frac{\alpha_1(k)}{\alpha_2(k)}$$

Selanjutnya bergantung pada parameter faktor pengabai $\alpha_1(k)$ dan $\alpha_2(k)$.

Untuk tipe variable faktor pengabai, parameter $\alpha_1(k)$ dan $\alpha_2(k)$ dibuat sebagai berikut :

$$\alpha_1(k) = \alpha_0 \alpha_1(k-1) + 1 - \alpha_0 \quad (2.16)$$

$$0 < \alpha_0 < 1 \text{ dan } \alpha_2(k) = 1$$

Dalam praktek diambil $\alpha_1(0) = 0,95 - 0,99$

Arti fisis dari algoritma rekursif penguatan adaptasi diatas adalah bobot pada faktor koreksi menurun asimptotik. Algoritma semacam ini dimaksudkan untuk mempercepat terjadinya konvergensi. Selanjutnya diperoleh estimasi parameter sistem :

$$\hat{\theta}(k) = \hat{\theta}(k-1) + F(k) \phi(k-1) \varepsilon(k) \quad (2.17)$$

2.3. JARINGAN SYARAF TIRUAN

Metode jaringan syaraf tiruan atau yang disebut neural network ini pada dasarnya diawali oleh pemikiran manusia untuk menciptakan suatu sistem yang mempunyai kemampuan mempelajari dan mengingat dari pengalaman yang pernah ditemui atau hal-hal yang pernah diterimanya.

Sifat belajar dan mengingat ini diilhami dari pengetahuan cara kerja sel otak manusia. Dilakukan penelitian bagaimana struktur yang terdapat pada otak dan cara kerja komponen-komponen penyusunnya sehingga otak mempunyai kemampuan mengingat dan mempelajari hal-hal yang pernah diberikannya.

Otak terdiri dari banyak sel penyusunnya. Pada sistem kerja otak, proses respon pada sel-selnya dilakukan secara paralel dan saling berhubungan. Pada tiap output suatu sel selalu dihubungkan ke input sel yang lain. Bagaimana dengan konfigurasi dan cara kerja seperti itu akhirnya mempunyai kemampuan dapat belajar dan mengingat apa yang pernah diterimanya, hal inilah yang mendorong ditemukannya beberapa metode untuk mengimplementasikan cara kerja tersebut sehingga didapatkan sistem yang mempunyai kemampuan seperti otak manusia.

Kemampuan pembelajaran dan mengingat dari pengalaman ini diarahkan kepada proses yang dapat dilakukan komputer agar dapat mempunyai kemampuan yang sama. Selama ini cara kerja komputer hanya bersifat sequensial sehingga diusahakan suatu metode yang dapat menjadikannya mampu melakukan suatu pekerjaan yang sebenarnya bersifat paralel, dengan tujuan akhir dapat mengimplementasikan sistem kerja otak sehingga sistem tersebut mempunyai kemampuan pembelajaran dan mengingat.

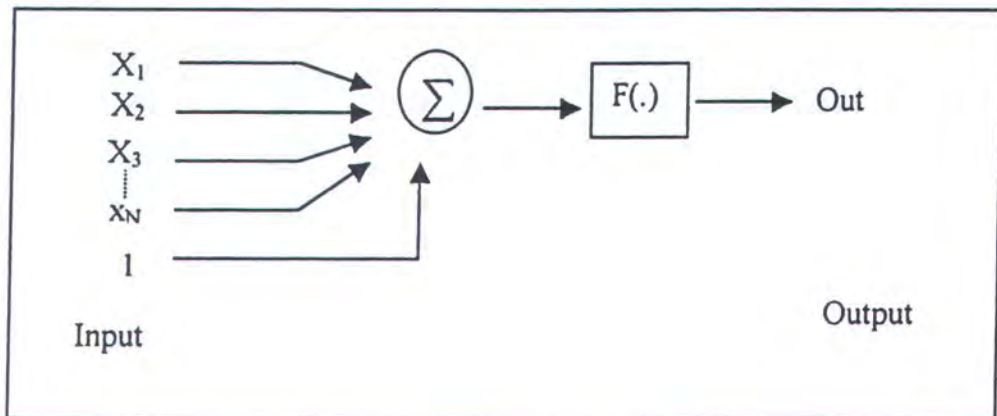
2.3.1 Pemodelan Jaringan syaraf tiruan

Didefinisikan pada gambar 2.1 merupakan unit neuron lapisan j yang diberi pola input x sebanyak i , dan bias $\theta = 1$, maka secara matematis output y berbentuk :

$$NET_j = \sum_{i=1}^N x_{ij} w_{ij} + \theta_{ji} \quad (2.18)$$

$$y = f(NE_j) \quad (2.19)$$

Dengan w adalah elemen-elemen pembobot input dan $f(.)$ merupakan fungsi keluaran jaringan.



Gambar 2.1 Model sel syaraf

Input x_i yang masuk ke neuron dapat berupa informasi dari luar sistem atau dari output lapisan neuron sebelumnya (lapisan neuron yang hirarkinya lebih rendah). Jika jaringan syaraf tersebut memiliki banyak lapisan neuron (multi layer), maka hanya lapisan neuron dengan hirarki yang paling rendah (lapisan input) yang menerima informasi eksternal.

Jaringan syaraf hanya dapat mengolah input bilangan. Jika masalah melibatkan besaran kualitatif atau pola gambar, maka input harus dikuantisasi ke bilangan yang bersesuaian sebelum diolah dengan jaringan syaraf.

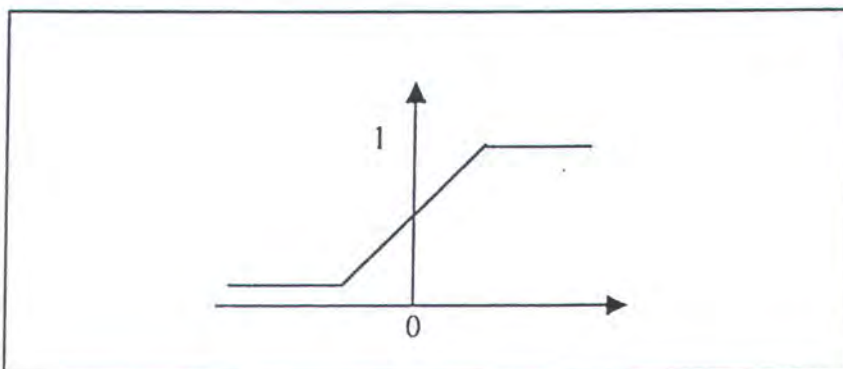
Elemen pembobot w pada setiap neuron merupakan representasi kekuatan sambungan (strength of synapse) antar neuron. Pembobot w_{ij} , pada model diatas, menyatakan kekuatan hubungan antara neuron-neuron lapisan i ke lapisan j . Jika neuron pada lapisan i berjumlah n , dan m pada lapisan j , maka pembobot w_{ij} berbentuk matriks $m \times n$. Penulisan subskrip pada w selalu dimulai dari lapisan hirarki tertinggi, sehingga untuk model tadi pembobot ditulis w_{ji} .

Semua data input yang telah diboboti dengan kekuatan sambungannya akan dijumlahkan terlebih dahulu sebelum diolah menjadi sinyal output. Hasil penjumlahan ini biasanya ditulis NET.

Fungsi yang mengolah data input menjadi data output disebut fungsi output atau aktuasi neuron $f(.)$. Fungsi ini biasanya berupa fungsi pemampat (squasing function), yang dapat berupa fungsi sigmoid

$$f(x) = \frac{1}{(1 + \exp(-x))} \quad (2.20)$$

berbentuk huruf S dan simetri terhadap nilai tertentu. Seperti ditunjukkan pada gambar 2.2.



Gambar 2.2 Kurva output fungsi sigmoid

Kebanyakan jaringan syaraf menggunakan elemen bias sebagai bagian dari tiap lapisan kecuali untuk lapisan output. Elemen ini biasanya diberi harga aktivasi konstan sama dengan 1, dihubungkan ketiap unit neuron pada lapisan berikutnya dan pembobotannya diatur selama proses pembelajaran. Elemen bias akan memberikan suatu harga konstan pada penjumlahan pembobot tiap neuron. Penambahan elemen bias ini kadang dapat mempercepat proses konvergensi dari jaringan. Elemen bias juga akan memberikan efek treshold pada tiap neuron yang terhubung. Harga konstan pada hasil penjumlahan NET, akan menggeser kurva sigmoid kekiri atau kekanan. Misalnya jaringan syaraf menggunakan fungsi sigmoid sebagai fungsi aktivasi output dan elemen bias seharga 1, dengan pembobotnya,

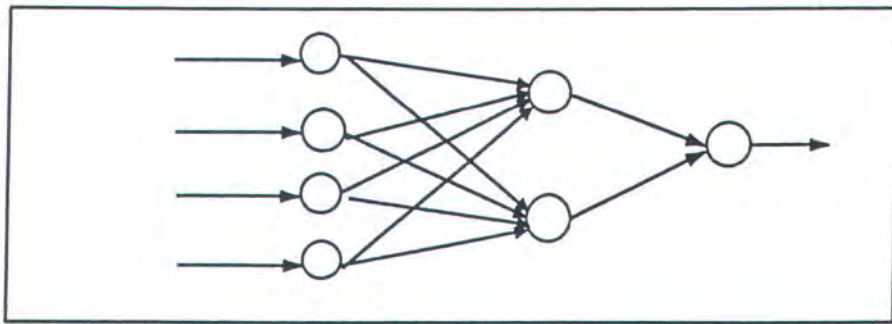
$$C = w_{j0} \quad (2.21)$$

Jika Z merupakan hasil penjumlahan semua input yang masuk pada neuron tersebut dan bias ini mempunyai kontribusi sebesar C pada input masukan unit j , maka penjumlahan ini menjadi :

$$Z + C \quad (2.22)$$

2.3.2 Arsitektur Jaringan Syaraf Tiruan

Arsitektur Jaringan Syaraf Tiruan yang digunakan pada penulisan ini adalah jaringan syaraf tiruan multi lapis (multi layer), yang memiliki lapisan hidden yang terletak antara lapisan input dengan lapisan output. Berdasar arah aliran informasinya, jaringan syaraf tiruan yang digunakan adalah jaringan umpan maju (feedforward network). Arsitektur jaringan syaraf tiruan seperti pada gambar berikut ini :



Gambar 2.3 Arsitektur jaringan syaraf tiruan

Jaringan umpan maju mempunyai struktur hirarki dimana antara neuron-neuron dalam satu lapisan dengan lapisan sebelumnya, tersambung dalam arah aliran informasi dari tingkat yang lebih rendah maju ke lapisan berikutnya atau yang hirarkinya lebih tinggi. Jadi antara output dengan input tidak terjadi aliran informasi yang terbalik atau umpan balik.

2.3.3 Algoritma Belajar

Pembelajaran (learning) bagi jaringan syaraf tiruan merupakan proses mengatur harga-harga dari parameter bobotnya untuk mendapatkan yang terbaik, dengan melatih (training) jaringan, menurut unjuk kerja sistem yang dikehendaki.

Tujuannya agar kumpulan pola input (vektor input) yang diberikan menghasilkan pola output (vektor output) yang diinginkan atau paling sedikit mendekati. Training ini dibentuk dengan menerapkan secara berurutan pola vektor input dengan mengatur pembobot jaringan agar mendekati pola vektor output, mengikuti suatu algoritma pembelajaran tertentu. Selama proses pembelajaran, pembobot secara perlahan konvergen menuju harga tertentu, sehingga pola input menghasilkan pola output yang diinginkan.

Algoritma belajar yang digunakan dalam penulisan ini adalah algoritma belajar secara supervised, yang memerlukan pasangan input dan output.

Metode belajar yang ada sekarang ini dikembangkan dari konsep yang diusulkan oleh Hebb, dimana pembobot akan ditambah jika neuron sumber dan tujuan diaktifkan. Hal ini analog dengan memperkuat lintasan-lintasan jaringan.

Secara matematis dapat dituliskan sebagai berikut :

$$w_{ji}(n+1) = w_{ji}(n) + \eta y_{ij} y_j \quad (2.23)$$

dimana $w_{ji}(n)$ adalah harga pembobot dari neuron i ke neuron j sebelum diatur, $w_{ji}(n+1)$ adalah harga pembobot dari neuron i ke neuron j sesudah diatur.

η adalah koefisien learning rate, y_{ij} adalah keluaran dari neuron i dan merupakan input dari neuron j , dan y_j adalah output dari neuron j .

2.3.4. Algoritma Back Propagation

Algoritma pembelajaran Back-propagation pertama diformulasikan oleh Werbos dan dipopulerkan oleh Rumelhart dan McClelland untuk dipakai pada jaringan syaraf, didisain untuk beroperasi pada jaringan feed forward multilapis dengan metode "supervised".

Algoritma ini banyak dipakai pada aplikasi pengendalian karena prosedur pembelajarannya didasarkan pada hubungan yang sederhana. Jika output memberikan hasil yang salah maka pembobot dikoreksi supaya error dapat diperkecil dan respon jaringan selanjutnya diharapkan akan mendekati harga yang benar. Prosedur ini analog dengan proses pengaturan pada sistem pengaturan. Back-propagation juga berkemampuan untuk menangani pelatihan pembobot pada lapisan

hidden. Tetapi kekurangan utama pada back-propagation adalah konvergensi hasil yang tidak dijamin, tetapi dengan perhitungan matematis yang sederhana algoritma ini tetap banyak dipakai.

Fungsi aktivasi output yang biasa dipergunakan adalah jenis fungsi sigmoid atau fungsi hiperbolik, karena metode back-propagation memerlukan jenis fungsi yang dapat diturunkan (differentiable) dan bersifat monoton non decreasing. Penambahan harga aktivasi hanya dapat menambah output atau tetap, tanpa pernah berkurang.

Penguatan pada fungsi ini merupakan kemiringan kurva pada suatu tingkat eksitasi tertentu, dan nilainya berubah dari harga yang kecil pada eksitasi negatif besar menjadi harga yang besar pada eksitasi nol dan kemudian kembali mengecil atau tetap pada harga eksitasi yang semakin besar dan positif.

2.3.5 Tahapan pada Algoritma Backpropagation

Algoritma back-propagation terdiri dari dua tahap yaitu tahapan propagasi maju (forward propagation) dan tahapan propagasi balik (backward propagation). Tahapan propagasi maju dimulai dengan memberikan suatu pola masukan pada lapisan input. Pada lapisan input ini, pola masukan tadi hanya dilewatkan untuk kemudian dikalikan dengan pembobot yang menghubungkannya dengan lapisan di atasnya (lapisan hidden). Lapisan input ini adalah merupakan lapisan pasif karena tidak mengolah pola masukan yang diterimanya. Dalam setiap lapisan yang berurutan (kecuali lapisan input), setiap elemen pengolah (neuron) menjumlahkan setiap masukannya dan melewatkannya pada fungsi aktivasi untuk mendapatkan

outputnya. Output ini kemudian didistribusikan maju ke lapisan berikutnya, untuk kemudian mengalami proses yang sama sampai pada lapisan output. Lapisan output lalu menghasilkan output jaringan secara keseluruhan.

Tahapan propagasi balik dimulai dengan membandingkan respon jaringan keseluruhan dengan pola output yang diinginkan. Jika terdapat perbedaan (error) maka error tersebut akan digunakan untuk memperbaiki harga pembobot jaringan, dengan mendistribusikannya kembali pada lapisan sebelumnya.

2.3.5.1. Propagasi Maju

Tahapan ini dimulai dengan memberikan suatu pola masukan pada jaringan.

Misalnya untuk satu vektor masukan berikut :

$$x_p = (x_{p1}, x_{p2}, \dots, x_{pN})^T \quad (2.24)$$

Tiap unit input akan mendistribusikan pola masukan tersebut ke neuron pada lapisan berikutnya (hidden). NET input untuk unit hidden ke $-j$ adalah sebagai berikut :

$$\text{net}_{pj}^h = \sum_{i=1}^N w_{ji}^h x_{pi} + \theta_j^h \quad (2.25)$$

Dimana w_{ji} adalah pembobot sambungan dan unit input i ke j , dan θ adalah bias. Subskrip h mengacu pada kuantitas pada lapisan tersembunyi (hidden). Anggaplah aktivasi dari mode ini sama dengan NET input, maka output dari node ini adalah sebagai berikut :

$$i_{pj} = f_j^h(\text{net}_{pj}^h) \quad (2.26)$$

Sehingga persamaan output menjadi :

$$\text{net}_{pk}^o = \sum_{j=1}^L w_{kj}^o i_{pj} + \theta_k^o \quad (2.27)$$

$$o_{pk} = f_k^o(\text{net}_{pk}^o) \quad (2.28)$$

Dengan subskrip o mengacu pada kuantitas pada lapisan output.

2.3.5.2. Propagasi Balik

Tahapan propagasi balik dimulai dengan membandingkan output jaringan keseluruhan dengan output yang diinginkan. Error yang terjadi digunakan untuk memperbaharui tiap pembobot jaringan.

Error ini diminimisasikan oleh persamaan berikut :

$$E_p = \frac{1}{2} \sum_{k=1}^M \delta_{pk}^2 \quad (2.29)$$

Dengan $\delta_{pk} = (y_{pk} - o_{pk})$ adalah error pada lapisan output.

Pergerakan harga E_p ini sering juga dipandang sebagai suatu permukaan pada ruang pembobot (weight space).

Arah perubahan bobot adalah gradien negatif dari E_p yang dihitung dengan memperhatikan harga bobot w_{kj} . Perubahan pembobot terjadi sesuai dengan arah gradien negatif. Kemudian harga pembobot dapat diatur secara iterative supaya E_p dapat berkurang atau mencapai minimum pada ruang pembobot tersebut.

Berikut ini penurunan mekanisme back-propagation dalam memperbaiki harga tiap pembobotnya. Persamaan diatas dapat ditulis sebagai berikut :

$$E_p = \frac{1}{2} \sum_{k=1}^M (y_{pk} - o_{pk})^2 \quad (2.30)$$

Maka gradien negatif dari E_p terhadap harga pembobot w_{kj} adalah :

$$\frac{\partial E_p}{\partial w_{kj}^o} = - (y_{pk} - o_{pk}) \frac{\partial f_k^o}{\partial (\text{net}_{pk}^o)} \frac{\partial (\text{net}_{pk}^o)}{\partial w_{kj}^o} \quad (2.31)$$

$$\frac{\partial (\text{net}_{pk}^o)}{\partial w_{kj}^o} = \left[\frac{\partial}{\partial w_{kj}^o} \sum_{j=1}^L w_{kj}^o i_{pj} + \theta_k^o \right] = i_{pj} \quad (2.32)$$

Sehingga diperoleh gradien negatif sebagai berikut :

$$\frac{\partial E_p}{\partial w_{kj}^o} = - (y_{pk} - o_{pk}) f_k^o (\text{net}_{pk}^o) i_{pj} \quad (2.33)$$

dengan

$$f_k^o (\text{net}_{pk}^o) = \frac{\partial f_k^o}{\partial (\text{net}_{pk}^o)} \quad (2.34)$$

2.3.6. Jaringan Pada Lapisan Output

Pembobot jaringan pada lapisan output diperbaharui mengikuti persamaan berikut :

$$w_{kj}^o (t + 1) = w_{kj}^o (t) + \Delta p w_{kj}^o (t) \quad (2.35)$$

dimana :

$$\Delta p w_{kj}^o = \eta (y_{pk} - o_{pk}) f_k^o (\text{net}_{pk}^o) i_{pj}$$

Dan η adalah konstanta belajar, yang biasanya berharga $0 < \eta < 1$

Dengan fungsi aktivasi output yang digunakan dalam menurunkan algoritma ini adalah fungsi sigmoid.

$$f_k^o (\text{net}_{pk}^o) = (1 + \exp^{-\text{net}_{pk}^o})^{-1} \quad (2.36)$$

Yang mempunyai fungsi turunan sebagai berikut :

$$f_k^{o'} = f_k^o (1 - f_k^o) = o_{pk} (1 - o_{pk}) \quad (2.37)$$

Maka persamaan untuk mengubah harga pembobot diatas menjadi sebagai berikut :

$$w_{kj}^o (t + 1) = w_{kj}^o (t) + \eta (y_{pk} - o_{pk}) o_{pk} (1 - o_{pk}) i_{pj} \quad (2.38)$$

Sehingga jika didefinisikan

$$\begin{aligned} \delta_{pk}^o &= (y_{pk} - o_{pk}) f_k^{o'} (net_{pk}^o) \\ &= \delta_{pk} f_k^{o'} (net_{pk}^o) \end{aligned} \quad (2.39)$$

Maka persamaan untuk memperbaharui harga pembobot dapat ditulis sebagai berikut :

$$w_{kj}^o (t + 1) = w_{kj}^o (t) + \eta \delta_{pk}^o i_{pj} \quad (2.40)$$

2.3.7. Jaringan Pada Lapisan Hidden (Tersembunyi)

Pembobot pada lapisan tersembunyi (hidden) diperbaiki mirip dengan pada lapisan output, namun dengan harga output target yang tidak diketahui. Secara intuitif, total error E_p dapat diketahui berhubungan dengan harga output pada lapisan ini, sehingga didapatkan

$$E_p = \frac{1}{2} \sum_{k=1} (y_{pk} - o_{pk})^2 \quad (2.41)$$

$$E_p = \frac{1}{2} \sum_{k=1} (y_{pk} - f_k^o (net_{pk}^o))^2 \quad (2.42)$$

$$E_p = \frac{1}{2} \sum_{k=1}^M \left(y_{pk} - f_k^o \left(\sum_j w_{kj}^o i_{pj} + \theta_k^o \right) \right)^2 \quad (2.43)$$

Telah diketahui bahwa i_{pj} tergantung pada pembobot lapisan hidden, sehingga gradien E_p dengan mengacu pada pembobot lapisan hidden dapat dihitung sebagai berikut :

$$\frac{\partial E_p}{\partial w_{ji}^h} = \frac{1}{2} \sum_k \frac{\partial}{\partial w_{ji}^h} - (y_{pk} - o_{pk})^2 \quad (2.44)$$

$$\frac{\partial E_p}{\partial w_{ji}^h} = - \sum_k (y_{pk} - o_{pk}) \frac{\partial o_{pk}^o}{\partial (\text{net}_{pk}^o)} \frac{\partial (\text{net}_{pk}^o)}{\partial i_{pj}} \frac{\partial i_{pj}}{\partial (\text{net}_{pj}^h)} \frac{\partial (\text{net}_{pj}^h)}{\partial w_{ji}^h} \quad (2.45)$$

Karena :

$$\frac{\partial o_{pk}^o}{\partial (\text{net}_{pk}^o)} = f_k^{o'}(\text{net}_{pk}^o)$$

$$\frac{\partial (\text{net}_{pk}^o)}{\partial i_{pj}} = w_{kj}^o$$

$$\frac{\partial i_{pj}}{\partial (\text{net}_{pj}^h)} = f_j^{h'}(\text{net}_{pj}^h)$$

$$\frac{\partial (\text{net}_{pj}^h)}{\partial w_{ji}^h} = x_{pi}$$

Maka persamaan diatas dapat ditulis sebagai berikut :

$$\frac{\partial E_p}{\partial w_{ji}^h} = - \sum_k (y_{pk} - o_{pk}) f_k^{o'}(\text{net}_{pk}^o) w_{kj}^o f_j^{h'}(\text{net}_{pj}^h) x_{pi} \quad (2.46)$$

Pembobot pada lapisan hidden diperbaharui sebanding dengan gradien negatif dari E_p , atau harga negatif dari persamaan berikut :

$$\Delta w_{ji}^h = \eta f_j^{h'}(\text{net}_{pj}^h) x_{pi} \sum_k (y_{pk} - o_{pk}) f_k^{o'}(\text{net}_{pk}^o) w_{kj}^o \quad (2.47)$$

Dengan η tetap sebagai learning rate, karena harga δ_{pk}^o telah diketahui, maka persamaan tersebut diatas dapat ditulis sebagai berikut :

$$\Delta w_{ji}^h = \eta f_j^{h'}(\text{net}_{pj}^h) x_{pi} \sum_k \delta_{pk}^o w_{kj}^o \quad (2.48)$$

Terlihat bahwa perbaikan setiap pembobot pada lapisan hidden tergantung pada harga error δ_{pk}^o dari lapisan output.

Adanya mekanisme inilah yang membuat algoritma ini juga dikenal dengan istilah error back-propagation. Mekanisme ini juga memperlihatkan kemampuannya untuk melatih lapisan hidden dari jaringan dengan menghindari keterbatasan jaringan dengan satu lapisan.

Dengan mendefinisikan error pada lapisan hidden sebagai berikut :

$$\delta_{pj}^h = f_j^{h'}(\text{net}_{pj}^h) \sum_k \delta_{pk}^o w_{kj}^o \quad (2.49)$$

Maka persamaan untuk mengubah harga pembobot pada lapisan hidden dapat ditulis sebagai berikut :

$$w_{ji}^h(t+1) = w_{ji}^h(t) + \eta \delta_{pj}^h x_{pi} \quad (2.50)$$

2.4. KONTROLER LOGIKA FUZZY

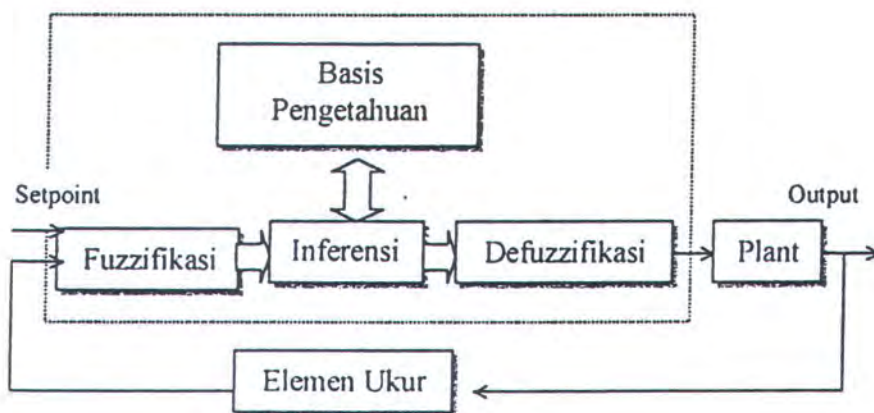
Perancangan Kontroler Logika Fuzzy (*Fuzzy Logic Controller*) atau KLF menggabungkan aspek pendefinisian himpunan fuzzy dengan aspek logika fuzzy untuk memperoleh suatu kontroler yang dapat merepresentasikan cara kerja operator manusia. Dengan prosedur perancangan tertentu kedua aspek diterapkan pada masukan dan keluaran untuk membentuk hasil perancangan yang berbentuk suatu algoritma aturan fuzzy.

2.4.1. Struktur Dasar KLF

Secara umum KLF mempunyai empat bagian pokok seperti yang ditunjukkan oleh Gambar 2.4.

Keempat bagian tersebut mempunyai fungsi sebagai berikut :

1. *Fuzzifier* berfungsi untuk mentransformasikan sinyal masukan yang bersifat *crisp* (bukan fuzzy) ke himpunan fuzzy dengan menggunakan operator *fuzzifier*.
2. *Basis Pengetahuan* (*Knowledge Base*) berisi basis data dan aturan dasar yang mendefinisikan himpunan fuzzy atas daerah-daerah masukan dan keluaran dan menyusunnya dalam perangkat aturan kontrol.
3. *Logika Pengambilan Keputusan* merupakan inti dari KLF yang mempunyai kemampuan seperti manusia dalam mengambil keputusan. Aksi atur fuzzy disimpulkan dengan menggunakan implikasi fuzzy dan mekanisme inferensi fuzzy.
4. *Defuzzier* berfungsi untuk mentransformasikan kesimpulan tentang aksi atur yang bersifat fuzzy menjadi sinyal sebenarnya yang bersifat *crisp* dengan menggunakan operator *defuzzier*.



Gambar 2.4 Struktur Dasar KLF

2.4.2. Fuzzifikasi

Dalam penerapan suatu sistem pengaturan, besaran masukan (referensi) dan besaran keluaran plant yang diperoleh melalui elemen ukur akan selalu berupa *crisp*

yang bersifat pasti dan kuantitatif, sedangkan pengolahan data dalam KLF didasarkan pada teori himpunan fuzzy yang menggunakan variabel linguistik yang bersifat fuzzy. Oleh karenanya pada tahap awal KLF diperlukan adanya fuzzifikasi yang dilakukan oleh *fuzzifier*. Fuzzifikasi sekaligus dapat dikatakan sebagai pemetaan yang memetakan masukan ke semesta himpunan fuzzy. Secara simbolis pemetaan ini dinyatakan dalam :

$$x = \text{fuzzifier}(x_0) \quad (2.51)$$

x_0 adalah masukan *crisp*, x adalah himpunan fuzzy dan *fuzzifier* merupakan operator fuzzifikasi.

Metode yang sering digunakan adalah dengan memperlakukan masukan *crisp* yang diperoleh sebagai suatu fuzzy tunggal (*fuzzy singleton*). Ini berarti x_0 dianggap sebagai himpunan fuzzy dengan fungsi keanggotaan $\mu(x) = 0$ kecuali pada titik x_0 yang nilai keanggotaannya $\mu(x) = 1$. Secara mendasar fuzzy tunggal adalah suatu nilai yang pasti dan bersifat bukan fuzzy. Jadi jelas dalam hal ini tidak ada kefuzzian dalam data yang dihasilkan.

2.4.3. Basis Data

Basis data berfungsi untuk mendefinisikan himpunan-himpunan fuzzy dari sinyal masukan dan sinyal keluaran agar dapat digunakan oleh variabel linguistik dalam aturan dasar. Perancangan basis data meliputi tiga pokok, yaitu :

- a. Kuantisasi dan normalisasi
- b. Pembagian ruang masukan dan keluaran
- c. Keluaran dan pemilihan fungsi keanggotaan

a. Kuantisasi dan Normalisasi

Kuantisasi berarti mendiskritkan semesta pembicaraan yang kontinyu ke dalam sejumlah segmen-segmen tertentu yang disebut level kuantisasi. Pemberian nomor atau label level-level ini membentuk pendukung himpunan fuzzy secara berhingga atau semesta pembicaraan baru yang bersifat diskrit. Prosedur ini diperlukan bila pendefinisian fungsi keanggotaannya dinyatakan secara numerik. Pendefinisian ini biasanya dinyatakan dalam bentuk tabulasi, yang penerapannya berbentuk suatu tabel pandang (*look-up table*).

Pemilihan jumlah level kuantisasi (pendukung) mempengaruhi kepekaan KLF terhadap masukan dan kehalusan aksi atur pada keluaran. Semakin banyak level kuantisasi yang diterapkan pada daerah masukan dan keluaran maka semakin peka KLF tersebut dan semakin halus aksi kontrolnya.

b. Pembagian Ruang Masukan dan Keluaran

Pendefinisian himpunan fuzzy atas daerah masukan dan keluaran berarti pula membagi-bagi semesta pembicaraan atas nilai-nilai variabel linguistik himpunan fuzzy. Nilai-nilai ini dinyatakan dengan seperangkat istilah linguistik seperti *NB*, *NS*, *NS*, *SN*, *PK*, *PS*, *PB* dan sebagainya.

Pembagian ruang masukan dan keluaran ini menentukan berapa banyak jumlah himpunan fuzzy dan aturan kontrolnya yang dapat disusun. Akibatnya secara langsung jumlah himpunan menentukan kehalusan pengaturan KLF.

c. Pemilihan Fungsi Keanggotaan

Pembagian ruang dilakukan dengan 7 nilai linguistik himpunan fuzzy yaitu : *NB*, *NS*, *NK*, *SN*, *PK*, *PM*, dan *PB*, sedangkan fungsi keanggotaannya dinyatakan :

$$\mu(x) = \sum \mu(x_i) \cdot x_i \quad (2.52)$$

Fungsi keanggotaan dapat dipilih secara bebas dengan menentukan secara sembarang nilai keanggotaannya. Tetapi pada dasarnya penentuan ini harus tetap dapat menggambarkan karakteristik masing-masing himpunan fuzzy.

Pada pendefinisian secara fungsional tidak diperlukan pemberian nomor level kuantisasi karena dalam hal ini tidak diperlukan pendukung yang bersifat diskrit. Pembagian ruang biasanya cukup dengan menentukan titik sumbu simetri dan sebaran (jangkauan) fungsi yang digunakan. Fungsi ini mudah diadaptasi terhadap keadaan semesta pembicaraan dengan cara mengatur parameter rata-rata yang menentukan titik kerja dan parameter sebaran yang menentukan jangkauan kerja.

2.4.4. Aturan Dasar

Sistem fuzzy dikarakteristikan oleh istilah-istilah linguistik dan biasanya berbentuk aturan-aturan jika-maka (*if-then*), yang disebut aturan kontrol fuzzy. Hal ini untuk mempermudah implementasi bentuk kondisional fuzzy. Seperangkat aturan kontrol fuzzy merupakan penjabaran dari bentuk-bentuk kondisional fuzzy dalam aturan dasar pada KLF. Perancangan perangkat aturan kontrol meliputi :

- a. Pemilihan variabel masukan dan keluaran.
- b. Penurunan aturan kontrol fuzzy.

a. Pemilihan Variabel Masukan dan Keluaran

Pemilihan variabel masukan dan keluaran memberi pengaruh kuat pada karakteristik kontroler. Dalam hal ini pengetahuan dan pengalaman rekayasa memainkan peranan yang sangat penting. Pemilihan variabel ini memerlukan

pemahaman perilaku *plant* dan perilaku pengaturannya.

Secara umum sebagaimana kontroler konvensional *error* dan perubahan *error* dari *plant* tetap menjadi acuan utama bagi aturan kontrol KLF untuk menghasilkan sinyal atur. KLF biasanya menggunakan variabel masukan berupa *error* dan perilaku *error* seperti perubahan *error*, turunan *error*, penjumlahan *error*, percepatan *error*, dan sebagainya, sedangkan keluaran atau aksi aturnya berupa sinyal atur atau perubahan sinyal atur.

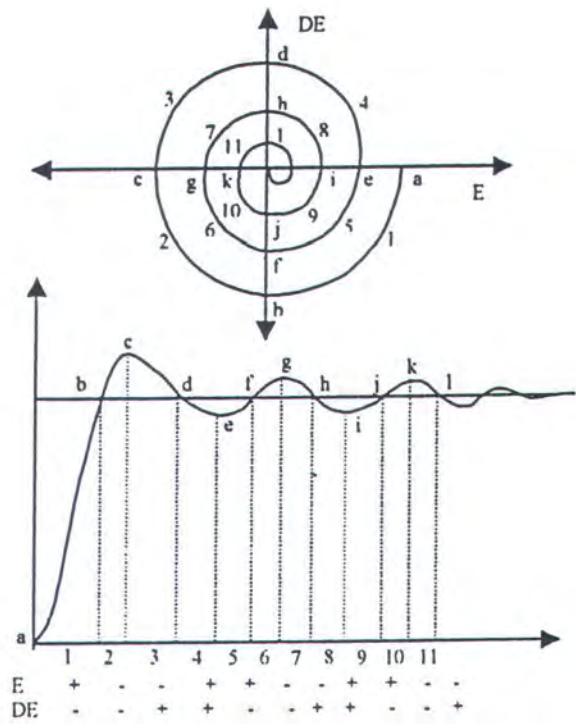
b. Penurunan Aturan Kontrol Kontrol Fuzzy

Salah satu cara menurunkan aturan kontrol fuzzy adalah dengan mengumpulkan aturan-aturan kontrol fuzzy yang dibentuk dari analisa perilaku obyek atur secara heuristik. Aturan kontrol diturunkan dengan jalan mengoreksi simpangan keluaran *plant* dari keadaan yang diinginkan. Penurunan benar-benar dilakukan secara heuristik dengan mengandalkan pengetahuan kualitatif. Hasil yang diperoleh akan berupa prototipe aturan kontrol secara garis besar. Sehingga hasilnya masih perlu disempurnakan.

Ada beberapa metode yang dapat digunakan untuk menyempurnakan prototipe aturan kontrol itu. King dan Mamdani (1977) mengusulkan metode penyempurnaan aturan kontrol dengan cara yang disebut Pemetaan Skala (*Scale Mappings*). Prinsipnya adalah menentukan sinyal atur sedemikian rupa sehingga trayektori sistem *loop* tertutup berakhir di keadaan yang diinginkan. Dalam hal ini pengetahuan tentang perilaku peralihan *plant*, seperti lewatan (*maximum overshoot*), waktu naik (*rise time*), dan sebagainya, serta intuisi tentang perilaku sistem tertutup diperlukan.

Gambar 2.5. menunjukkan tanggapan *plant loop* tertutup terhadap variabel masukan *plant*. Dalam hal ini ada dua variabel masukan KLF yang dapat digunakan, yaitu *error* (*e*) dan perubahan *error* (Δe). Untuk memperjelas trayektori dapat digunakan grafik yang menyatakan hubungan antara *error* (*e*) dan perubahan *error* (Δe), sedangkan variabel keluaran KLF berupa perubahan sinyal atur (Δu) atau sinyal atur (*u*).

Pada penurunan prototipe ini pembagian ruang masukan adalah 3, yang berarti ada 3 nilai linguistik yaitu positif (P), negatif (N) dan nol (SN).



Gambar 2.5. Tanggapan Sistem Loop Tertutup

Peninjauan kualitatif pertitik (bertanda huruf) pada grafik memberikan penalaran yang menghasilkan prototipe aturan kontrol pada Tabel 2.1

Tabel 2.1. Prototipe Aturan Kontrol Linguistik dengan 3 Nilai

Aturan No.	e	Δe	u	Titik Referensi
1	P	SN	P	a,e,i
2	SN	N	N	b,f,j
3	N	SN	N	c,g,k
4	SN	P	P	d,h,l
5	SN	SN	SN	set point

Tabel 2.2. Penyempurnaan Aturan Kontrol dengan 3 Nilai

Aturan No.	e	Δe	u	Titik Referensi
6	P	N	P	1(<i>rise time</i>),5
7	N	N	N	2(<i>overshoot</i>),6
8	N	P	N	3,7
9	P	P	P	4,8
10	P	N	SN	9
11	N	P	SN	10

Sedang peninjauan secara kualitatif per daerah (bertanda angka) pada grafik tanggapan maupun pada bidang fasa menunjukkan bahwa daerah 1 mempunyai pengaruh pada pemendekan waktu naik (*rise time*), dan daerah 2 berhubungan dengan pengurangan lewatan (*overshoot*), secara heuristik penalaran ini memberikan penambahan atau penyempurnaan aturan kontrol seperti pada Tabel 2.2.

Hasil yang diperoleh dari penyempurnaan aturan kontrol tersebut dapat dibuat aturan aturan kontrol yang lengkap untuk sistem MISO. MacVicar-Whelan yang meneliti masalah penyempurnaan aturan kontrol ini menemukan pola umum hubungan antara e , Δe dan u atau Δu . Penurunan secara lengkap aturan kontrol ini memberikan tabel keputusan sinyal atur yang disebut Tabel Keputusan MacVicar-Whelan. Tabel ini memberikan jumlah aturan kontrol secara maksimum. Pola ini berlaku bagi KLF yang menggunakan masukan *error* dan perubahan *error*. Tabel keputusan Mac Vicar –Whelan ditunjukkan pada Tabel 2.3.

Tabel 2.3. Tabel Keputusan Aturan Kontrol Fuzzy MacVicar-Whelan

		PERUBAHAN ERROR							
			NB	NS	NK	NL	PK	PS	PB
E R R O R	NB	NB	NB	NB	NB	NS	NK	NL	
	NS	NB	NB	NB	NS	NK	NL	PK	
	NK	NB	NB	NS	NK	NL	PK	PS	
	NL	NB	NS	NK	NL	PK	PS	PB	
	PK	NS	NK	NL	PK	PS	PB	PB	
	PS	NK	NL	PK	PS	PB	PB	PB	
	PB	NL	PK	PS	PB	PB	PB	PB	

2.4.5. Logika Pengambilan Keputusan

Berbagai cara telah digunakan untuk menentukan aturan mana yang digunakan berdasar dari nilai input dalam penentuan aksi kontrol yang sesuai. Ada dua metode *reasoning* yang sering digunakan saat ini yaitu¹ :

- Metode MAX – MIN *fuzzy reasoning*

¹ Jun Yan, Michael Ryan, James Power , Using Fuzzy Logic, Prentice Hall, 1994, p. 54

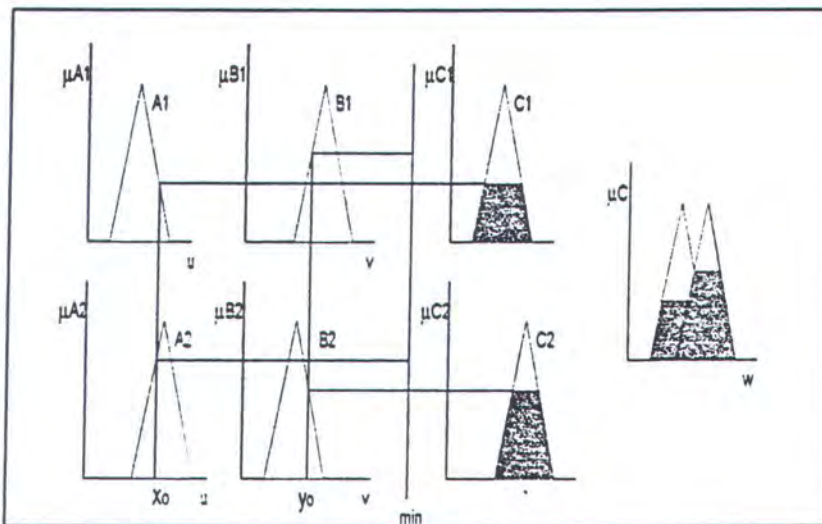
- Metode MAX – DOT *fuzzy reasoning*
- Pada penulisan ini digunakan metode Max min fuzzy reasoning

MAX-MIN fuzzy reasoning

Dalam MAX-MIN *fuzzy reasoning* aturan operasi minimum Mamdani R_c digunakan untuk implikasi fuzzy.

$$\mu_C(w) = (\alpha_1 \wedge \mu_{C1}(w)) \vee (\alpha_2 \wedge \mu_{C2}(w)) \quad (2.53)$$

Gambar 2.6 menunjukkan proses *inference* MAX-MIN untuk input *crisp* x_0 dan y_0 yang dianggap sebagai *fuzzy singletons*.



Gambar 2.6. MAX-MIN *fuzzy reasoning*

2.4.6. Defuzzifikasi

Pada dasarnya defuzzifikasi adalah pemetaan ruang aksi kontrol fuzzy menjadi ruang aksi kontrol non-fuzzy (*crisp*). Secara garis besar strategi defuzzifikasi bertujuan menghasilkan sinyal atur yang nyata yang dapat merepresentasikan distribusi dari aksi atur masing-masing aturan kontrol.

Bentuk umum proses defuzzifikasi adalah :

$$z_0 = \text{defuzzier}(z) \quad (2.54)$$

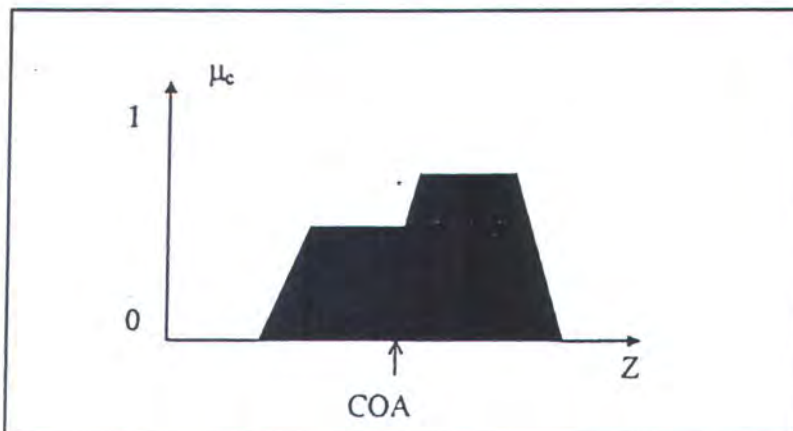
Pada beberapa metode defuzzifikasi yang dapat digunakan. Yang paling umum ada tiga macam yaitu kriteria max (*The Criterion Max*), titik tengah maksimum (*The Mean of Maximum*) dan metode titik pusat (*The Center of Area*). Pada penulisan ini digunakan metode center of area.

Metode Titik Pusat (COA)

Metode ini paling sering digunakan. Strategi COA dibangkitkan dari pusat berat (*the center of gravity*) pada aksi kontrol. Dalam kasus diskrit aksi kontrol ditunjukkan sebagai :

$$z_0 = \frac{\sum_{j=1}^n \mu_z(w_j) \cdot w_j}{\sum_{j=1}^n \mu_z(w_j)} \quad (2.55)$$

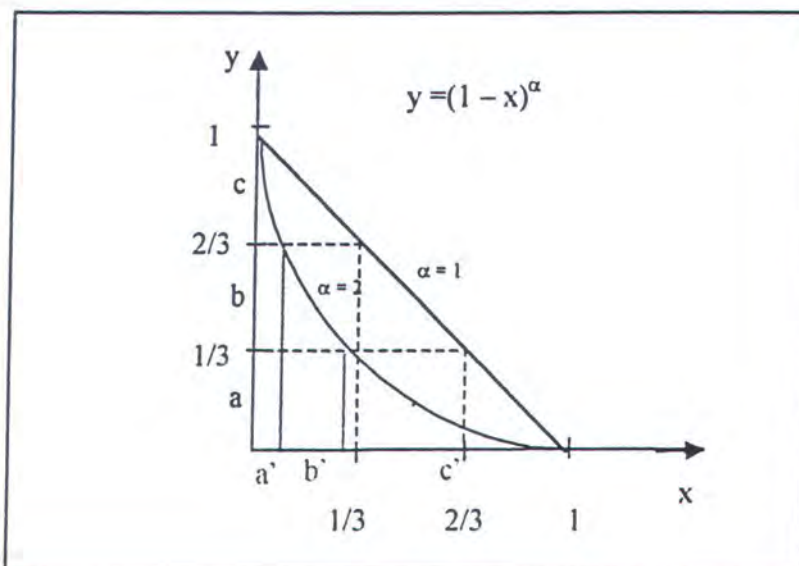
di mana n adalah banyaknya level kuantisasi dari keluaran.



Gambar 2.7. Interpretasi Grafik dari Strategi Defuzzifikasi

2.4.7. Perubahan range fungsi keanggotaan dengan Fungsi Koridor

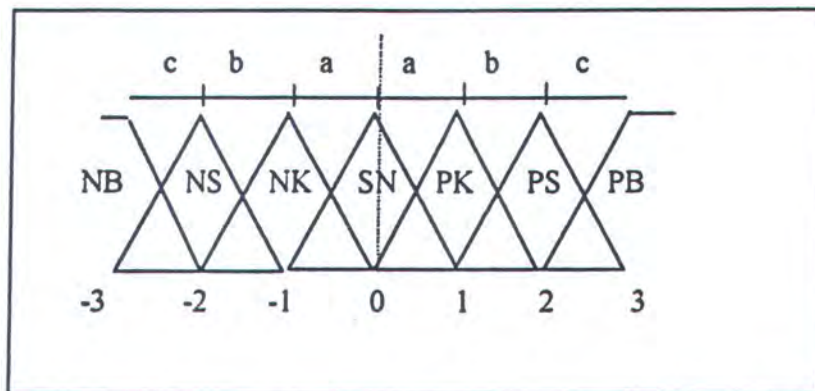
Performansi sistem pada kontroler logika fuzzy dapat diperbaiki dengan pengaturan rule-rule base dan pengaturan fungsi keanggotaan. Dalam hal ini dilakukan pengaturan fungsi keanggotaan himpunan fuzzy sehingga diperoleh output sistem yang lebih baik, yaitu sesuai dengan performansi sistem yang diinginkan. Salah satu cara yang digunakan dalam memodifikasi fungsi keanggotaan adalah dengan fungsi koridor. Fungsi koridor adalah suatu fungsi yang dapat digunakan untuk mengubah range tiap-tiap variable linguistik dalam memodifikasi fungsi keanggotaan. Misalnya jika dipilih fungsi koridor $y = (1 - x)^\alpha$, maka akan diperoleh titik-titik untuk berbagai nilai alfa (α), seperti pada gambar dibawah ini :



Gambar 2.8. Perubahan range fungsi keanggotaan dengan fungsi koridor

Dengan perubahan nilai alfa maka nilai-nilai $a : b : c$ juga akan berubah yang sekaligus digunakan untuk mengatur nilai fungsi keanggotaan. Pengaturan fungsi

keanggotaan dilakukan dengan mengubah range dari tiap-tiap variable linguistik dari suatu fungsi keanggotaan uniform seperti gambar 2.9.

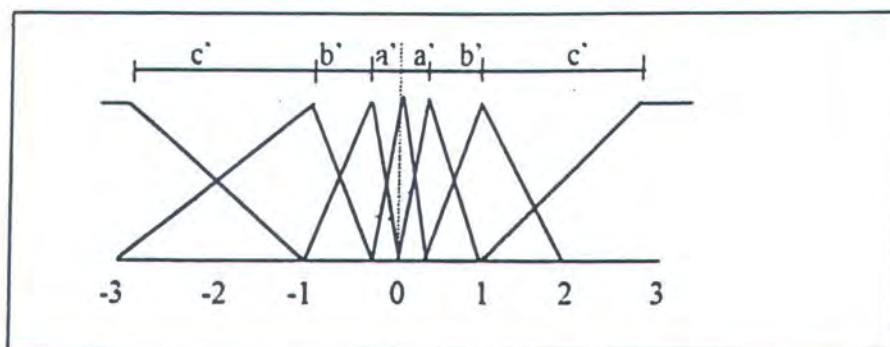


Gambar 2.9 Fungsi keanggotaan uniform ($\alpha = 1$)

Pada gambar diatas menunjukkan suatu fungsi keanggotaan uniform dimana range tiap-tiap variable linguistiknya dipengaruhi oleh nilai a, b dan c adalah sama untuk alpha sama dengan 1. Aturan untuk pengubahan fungsi keanggotaan pada penulisan ini adalah kaki kanan bawah dari suatu variable linguistik sama dengan puncak dari variable linguistik yang selanjutnya, sebagai contoh pada gambar 2.9 diatas pada $x = 0$, kaki kanan bawah untuk variable linguistik NK adalah puncak dari variable linguistik SN.

Satu contoh pengaturan fungsi keanggotaan Kontroler Logika Fuzzy adalah dengan mengubah nilai alpha maka nilai a : b : c juga berubah. Jadi untuk alpha sama dengan dua sesuai dengan fungsi koridor maka akan menghasilkan gambar seperti berikut :





Gambar 2.10 Fungsi keanggotaan yang telah dimodifikasi ($\alpha = 2$)

Pada gambar terlihat bahwa perubahan nilai perbandingan range tiap-tiap variable linguistik dipengaruhi oleh nilai $a : b : c$. Perubahan nilai perbandingan $a : b : c$ tersebut dapat diperoleh dari fungsi koridor, yaitu suatu fungsi yang memiliki nilai-nilai yang berubah seperti membuka dan menutupnya koridor suatu pintu yang berasal dari perubahan parameter pada bentuk dan fungsi-fungsi tertentu.

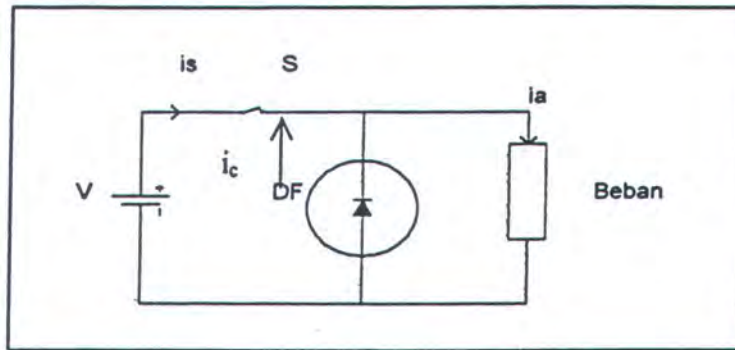
Pada penulisan ini fungsi koridor hanya dibatasi dengan bentuk segitiga yaitu dengan perubahan parameter yaitu α . Pembatasan ini dimaksudkan untuk penekanan pembahasan pada bagian pengaturan fungsi keanggotaannya dan pemilihan fungsi koridor ini telah mewakili secara heuristik bagi fungsi koridor yang berasal dari bentuk-bentuk dan fungsi-fungsi yang lainnya.

2.5. DC-DC KONVERTER

DC-DC Konverter digunakan untuk mengontrol motor dc sebab terdapat beberapa keuntungan seperti efisiensi yang tinggi, fleksibilitas dalam kontrol, ringan, ukuran kecil, dan responnya cepat. Dc-dc konverter disebut juga dengan nama DC chopper dan digunakan sebagai pengendali dc yang dapat digunakan dengan sumber

battery seperti forklift trucks, trolley juga digunakan sebagai pengendali motor dc penguat terpisah sebab memiliki karakteristik kontrol yang fleksible.

2.5.1 Prinsip operasi dan teknik kontrol dc-dc konverter



Gambar 2.11 Dasar rangkaian dc-dc konverter step down (klas A)

Diagram rangkaian dan gelombang steady state dari sebuah dc-dc konverter ditunjukkan didalam gambar 2.11. Sebuah sumber tegangan searah V menyuplai sebuah beban induktif melalui sebuah self-komutasi- semikonduktor saklar S . Simbol saklar S digunakan sebab sebuah dc-dc konverter dapat dibangun dengan menggunakan thyristor, GTO, transistor dan MOSFET. Dioda menunjukkan arah yang dapat dilewati arus. Sebuah dioda D_F dihubungkan secara paralel dengan beban. Semikonduktor switch S dioperasikan secara periodik dengan perioda T selama waktu $t_{on} = \delta T$, dengan $0 < \delta < 1$. Variable $\delta (= t_{on}/T)$ disebut duty ratio atau duty cycle dari sebuah dc-dc konverter. Dioda D_F menyediakan sebuah lintasan untuk aliran arus beban ketika saklar off sehingga memperbaiki gelombang arus beban. Dengan mempertahankan kekontinuan dari arus beban pada saat off, pada dasarnya adalah mempertahankan tegangan transient yang muncul melintasi saklar S . Komponen searah atau nilai rata-rata tegangan beban V_a diberikan oleh :

$$V_a = \frac{1}{T} \int_0^T V dt = \frac{1}{T} \int_0^{\delta T} V dt = \frac{t_{on}}{T} V = \delta V \quad (2.56)$$

Karena δ duty cycle dapat divariasikan dari 0 dan 1, dengan bervariasi menurut t_{on} , T atau f . Maka tegangan keluaran V_a dapat divariasikan dari 0 sampai V dengan mengatur δ .

Switch S dapat dikontrol dengan berbagai cara untuk duty ratio yang berubah-ubah $\delta(t_{on}/T)$. Teknik kontrol tersebut adalah:

1. Operasi pada frekuensi konstan. Frekuensi chopping f (atau periode chopping T) dijaga tetap dan waktu on, t_1 divariasikan. Lebar pulsa bervariasi dan kontrol jenis ini dikenal dengan nama kontrol pulse-width-modulation (PWM).
2. Operasi pada frekuensi variable. Frekuensi chopping bervariasi, sedang keadaan lainnya bias memilih kondisi yaitu waktu t_{on} dibuat konstan atau t_{off} dibuat konstan.

Karena apabila frekuensi dibuat bervariasi akan sulit dalam pembuatan filter dan mungkin terjadi ketidakkontinuan pada arus beban motor DC. Sehingga dengan melihat kerugian-kerugian diatas maka dc-dc konverter yang digunakan kebanyakan menggunakan sistem frekuensi konstan.

2.6. MOTOR DC PENGUAT TERPISAH

Motor DC merupakan suatu komponen yang mengubah energi listrik menjadi energi mekanik. Dasar kerja dari komponen ini adalah sebagai berikut : Suatu kumparan jangkar yang dialiri arus listrik i dalam suatu medan magnet B , maka akan terbangkit gaya sebesar :

$$F = B \cdot i \cdot l \quad (2.57)$$

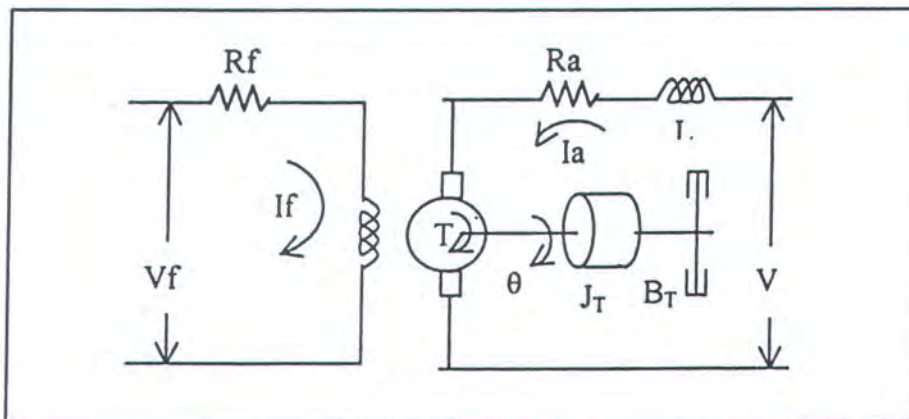
sehingga menghasilkan torsi sebesar :

$$T = B \cdot i \cdot l \cdot r \quad (2.58)$$

dengan r menunjukkan panjang *rotor*.

Torsi akan memutar *rotor* bila torsi yang terbangkit telah memiliki torsi lawan dari motor dan beban.

Motor DC yang dipakai dalam simulasi ini adalah motor DC penguat terpisah dengan medan *stator* yang dibangkitkan sumber tegangan. Dalam hal ini diharapkan arus jangkar yang dihasilkan dapat menghasilkan torsi yang diharapkan supaya menghasilkan output motor yang mendekati setpoint. Secara prinsip motor DC penguat terpisah dapat digambarkan seperti Gambar 2.12.



Gambar 2.12. Model Motor DC dengan penguatan terpisah

Besarnya torsi motor dirumuskan secara matematik ditunjukkan oleh persamaan 259

$$T_m = K_m \cdot I_a \cdot I_f \quad (2.59)$$

di mana:

T_m = Torsi motor, Nm

K_m = Konstanta motor

I_a = Arus jangkar, ampere

I_f = Arus medan, ampere

R_f = Tahanan kumparan medan , ohm

L_f = Induktansi kumparan medan , henry

i_f = Arus kumparan medan , ampere

V_f = Tegangan medan , volt

R_a = Tahanan jangkar , ohm

L_a = Induktansi jangkar , henry

i_a = Arus jangkar , ampere

V_a = Tegangan jangkar , volt

θ = Perpindahan sudut poros motor , radian

T_m = Torsi yang dibangkitkan oleh poros motor , N-m

J_T = Momen inersia dari motor dan beban , kg-det²

B_T = Koefisien gesekan viskos dari motor dan beban , N-m/rad/det

Pada simulasi ini dipilih nilai – nilai untuk tiap komponen dan konstanta yang digunakan sebagai berikut :

r_f = 20 ohm

L_f = 10 henry

R_a = 2 ohm

L_a = 1 henry

J_m = $2 \cdot 10^{-5}$ kg-det²

$$J_l = 1 \cdot 10^{-5} \text{ kg-det}^2$$

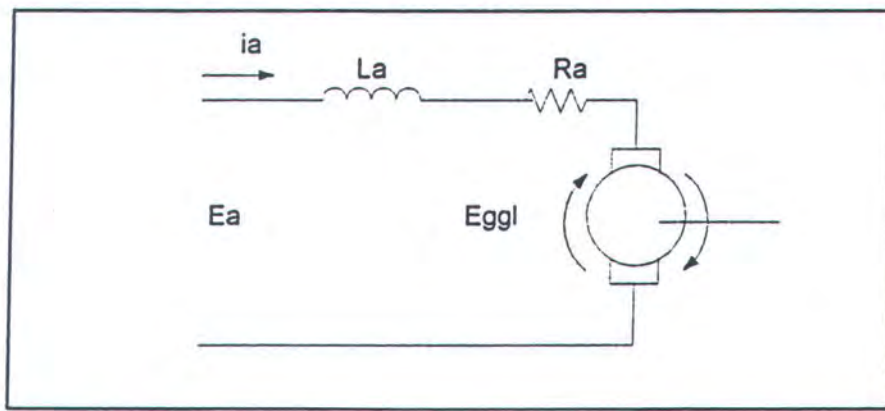
$$B_m = 1 \cdot 10^{-5} \text{ N-m/rad/det}$$

$$B_l = 1 \cdot 10^{-5} \text{ N-m/rad/det}$$

$$K_m = 1.2 \cdot 10^{-5} \text{ N-m/amp}$$

$$K_b = 0.1$$

Dari gambar 2.13 didapatkan rangkaian jangkar motor dc sebagai berikut :

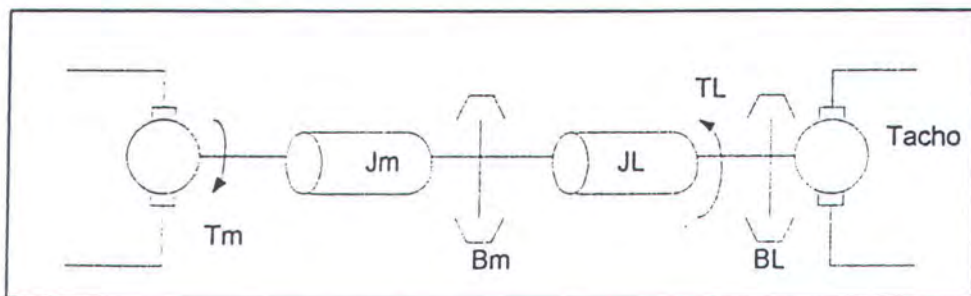


Gambar 2.13 Rangkaian jangkar motor dc

Dari gambar 2.13 didapatkan persamaan arus jangkar :

$$E_a(s) - E_{agl}(s) = (L_a s + R_a) I_a(s) \quad (2.60)$$

$$I_a(s) = \frac{1}{L_a s + R_a} (E_a(s) - E_{agl}(s)) \quad (2.61)$$



Gambar 2.14 Hubungan torsi dengan beban

Dari gambar 2.14 didapatkan persamaan :

Untuk persamaan torsi :

$$T_m = K_m \cdot I_a \cdot I_f \quad (2.62)$$

Untuk persamaan beban :

$$T_L = J_L \frac{d\omega_m}{dt} + B_L \omega_m \quad (2.63)$$

$$T_m - T_L = J_m \frac{d\omega_m}{dt} + B_m \omega_m \quad (2.64)$$

$$T_m = (J_m + J_L) \frac{d\omega_m}{dt} + (B_m + B_L) \omega_m \quad (2.65)$$

dimana :

$$J_m + J_L = J_T$$

$$B_m + B_L = B_T$$

maka :

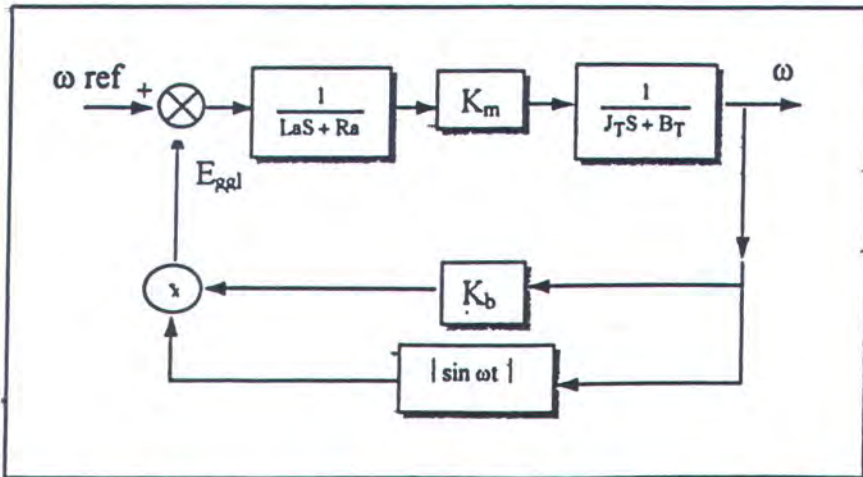
$$T_m(s) = (J_T s + B_T) \omega(s) \quad (2.66)$$

$$\omega(s) = \frac{1}{(J_T s + B_T)} T_m(s) \quad (2.67)$$

Dari Gambar 2.13 dapat dilihat bahwa kumparan jangkar berputar dengan kecepatan ω menggerakkan beban. Rangkaian jangkar terdiri dari resistansi jangkar R_a dan induktansi jangkar L_a , keduanya menjadi satu dalam kumparan jangkar. Tegangan yang menginduksi kumparan jangkar timbul akibat berputarnya kumparan jangkar memotong garis gaya magnetik, gaya tegangan ini besarnya E_{ggl} . Tegangan ini dinamakan *Gaya Gerak Listrik (GGL)*. E_{ggl} sebanding dengan kecepatan putar motor ω dengan pembanding K_b (konstanta GGL) sehingga dapat dituliskan :

$$E_{ggl} = K_b \cdot I_f \cdot \omega \cdot | \sin \omega t | \quad (2.68)$$

di mana K_b dalam ohm/rad ; ω dalam rad/det.



Gambar 2.15 Blok diagram plant motor DC dengan pengaturan jangkar

2.7. SPESIFIKASI KONTROL

Dalam setiap perancangan kontroler, diperlukan suatu spesifikasi kontrol yang akan digunakan sebagai acuan perancangan. Pada perancangan kontroler untuk pengaturan kecepatan motor dc dengan mengatur tegangan jangkar dan medan ini, digunakan spesifikasi performansi untuk respon transien sebagai acuan untuk mendapatkan suatu kontroler yang baik.

Karakteristik respon waktu (Time Respons) adalah karakteristik respon yang spesifikasi performansinya didasarkan pada pengamatan bentuk respon output sistem terhadap berubahnya waktu. Secara umum spesifikasi performansi respon waktu dapat dibagi atas dua tahapan pengamatan, yaitu :

- Spesifikasi Respons Transient, adalah spesifikasi respons sistim yang diamati mulai saat terjadinya perubahan sinyal input/gangguan/beban sampai respons masuk dalam keadaan steady state. Tolok ukur yang digunakan untuk mengukur

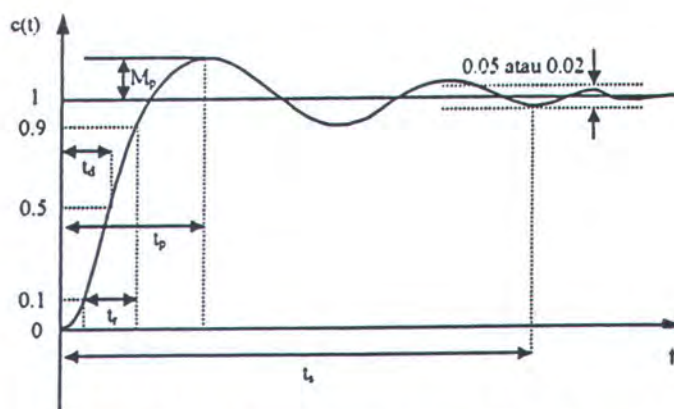
kualitas respon transient ini antara lain; Rise Time, Delay Time, Peak Time, Settling Time, %Overshoot.

- Spesifikasi Respons Steady State, adalah spesifikasi respon sistem yang diamati mulai saat respon masuk dalam keadaan steady state sampai waktu tak terbatas (dalam praktek waktu pengamatan dilakukan saat $T_s \leq t \leq 5T_s$). Tolok ukur yang digunakan untuk mengukur kualitas respons Steady state ini antara lain; % Error Steady State baik untuk error posisi, error kecepatan maupun error percepatan.

Parameter-parameter yang diperlukan untuk menganalisa karakteristik performansi sistem kontrol adalah sebagai berikut :

1. **Time Constant (τ)** : adalah waktu yang diukur melalui respon fungsi selubung yaitu mulai $t = 0$ s/d respons mencapai 63.2% dari respon steady state.
2. **Delay Time (T_d)** : adalah waktu yang diperlukan respon mencapai setengah harga akhir yang pertama.
3. **Rise Time (T_r)** : adalah waktu yang untuk respon naik dari 10 % sampai 90%, 5% sampai 95% atau 0% sampai 100% dari harga akhirnya.
4. **Overshoot (M_p)** : Nilai relatif yang menyatakan perbandingan harga maximum respons yang melampaui harga steady state dibanding dengan nilai steady state.
5. **Time Peak (T_p)** : adalah waktu untuk mencapai puncak lewatan (Overshoot) yang pertama.
6. **Settling Time (T_s)** : adalah waktu untuk menyatakan respons mencapai dan menetap pada $\pm 5\%$ atau $\pm 2\%$ atau $\pm 0.5\%$ dari respons Steady State.

demikian kurva respon transien secara virtual dapat diperoleh dengan menentukan harga-harga t_d , t_p , t_s , dan M_p , yang dapat dilihat pada Gambar 2.16.



Gambar 2.16. Spesifikasi Respon Transien

“...Allah akan meninggikan orang-orang diantaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat...”
(Al Mujaadilah : 11)

BAB III METODOLOGI PENELITIAN

BAB III

METODOLOGI PENELITIAN

Pada penulisan ini dikembangkan suatu metode pengendalian dengan mengkombinasikan sistem kontrol logika fuzzy dan jaringan syaraf tiruan yang bersifat adaptif untuk menghasilkan sinyal kontrol sebagai input bagi dc-dc konverter untuk mengatur kecepatan putaran dari motor dc penguat terpisah sesuai dengan yang performansi yang diinginkan. Disamping itu untuk melihat sejauh mana kontroler dapat mengantisipasi jika terjadi perubahan beban. Semuanya dilakukan dengan simulasi menggunakan program MATLAB 5.3.

3.1. LANGKAH-LANGKAH PENELITIAN

3.1.1. Pemodelan plant

Pada penelitian ini pertama-tama dilakukan studi literatur tentang parameter-parameter motor dc dan dc-dc konverter, yang sekaligus menjadi plant yang akan diatur nantinya oleh kontroler Fuzzy Adaptif. Dengan model transfer function yang telah didapatkan, akan dibuat simulasinya pada sistem open loop untuk melihat respon yang terbaik sesuai dengan spesifikasi respon yang diinginkan jika diberi beban nominal, beban minimum dan beban maximum. Pengaturannya dapat dilakukan dengan mengubah-ubah harga periode chopping.

3.1.2. Identifikasi parameter sistem plant

Identifikasi parameter dilakukan dengan metode off-line menggunakan teknik rekursif least square untuk memperoleh vektor parameter (θ) yaitu nilai-nilai a_1 , a_2 ,

... a_{nA} dan b_1, b_2, \dots, b_{nB} untuk berbagai perubahan beban. Jika ada sepuluh perubahan kombinasi perubahan beban maka akan ada sepuluh juga parameter a_1, a_2, b_0 dan b_1 . Pada penelitian ini perubahan beban dimisalkan sebagai perubahan torsi motor. Karena sistem dianggap deterministik dengan plant berorde dua dengan faktor delay satu, sehingga sistem didekati dengan model ARMA (Auto Regressive Moving Average) untuk $nA = 2$ dan $nB = 2$. Bentuknya menjadi :

$$Y(k) = -a_1 y(k-1) - a_2 y(k-2) + b_1 x(k-2) + b_2 x(k-3)$$

Adapun langkah-langkah iterasi metode least square :

- Baca data output sistem $y(t)$ dan $u(t)$
- Rekonstruksi vektor informasi sistem $\phi(t-1)$
- Menghitung kesalahan prediksi output apriori yaitu dengan formulasi :

$$\varepsilon_o(t) = y(t) - \phi^T(t-1)\hat{\theta}(t-1)$$

- Estimasi parameter $\hat{\theta}(t)$ dengan formulasi :

$$\hat{\theta}(t) = \hat{\theta}(t-1) + F(t) \phi(t-1) \varepsilon_o(t)$$

- Hitung faktor koreksi (gain adaptasi) berikutnya dengan formulasi :

$$F(t+1) = F(t) - \frac{F(t)\phi^T(t)\phi(t)F(t)}{1 + \phi^T(t)F(t)\phi(t)}$$

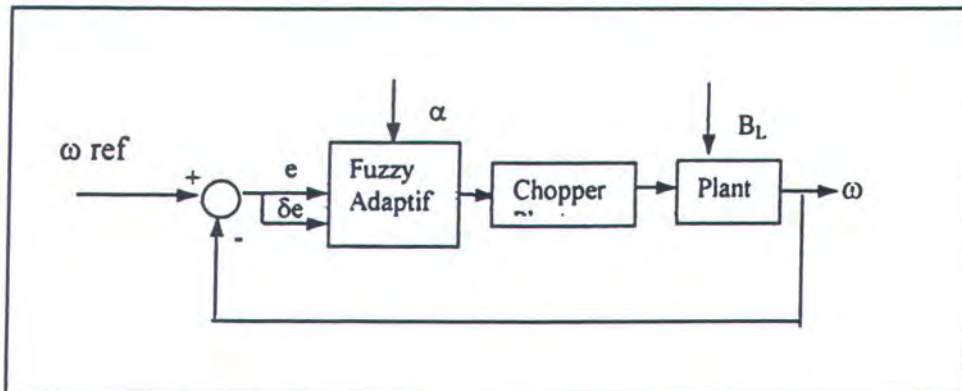
- Ulangi langkah 1 untuk $t = t + 2$

Pada langkah permulaan, yaitu $t = 0$, dapat diambil harga awal sebagai berikut :

$$\theta(0) = 0 \quad : \quad \phi(t-1) = 0 \quad : \quad F(0) = \alpha I, \quad \alpha > 0$$

3.1.3. Mencari nilai alfa (α) untuk tiap perubahan beban

Sesuai dengan spesifikasi respon yang diinginkan maka dicari nilai alfa untuk tiap perubahan beban. Blok diagram rangkaiannya sebagai berikut :

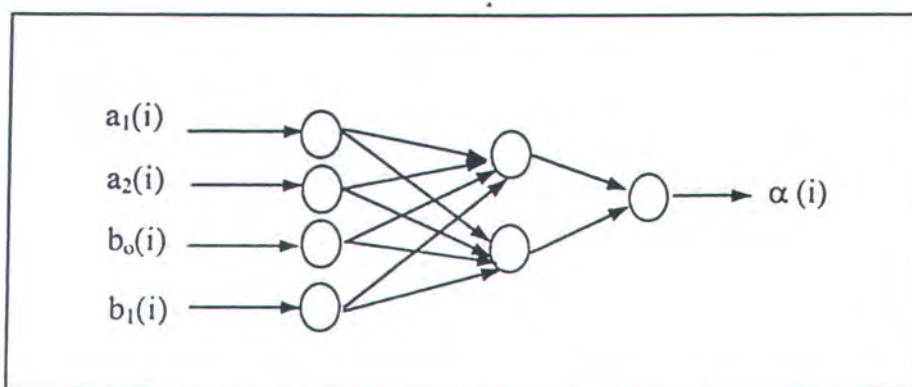


Gambar 3.1 Blok diagram untuk mencari nilai alfa

Sehingga untuk setiap perubahan beban yang sama dapat diperoleh nilai alfa.

Jadi jika ada sepuluh jenis perubahan beban maka juga terdapat sepuluh nilai alfa.

3.1.4. Pembelajaran jaringan syaraf tiruan

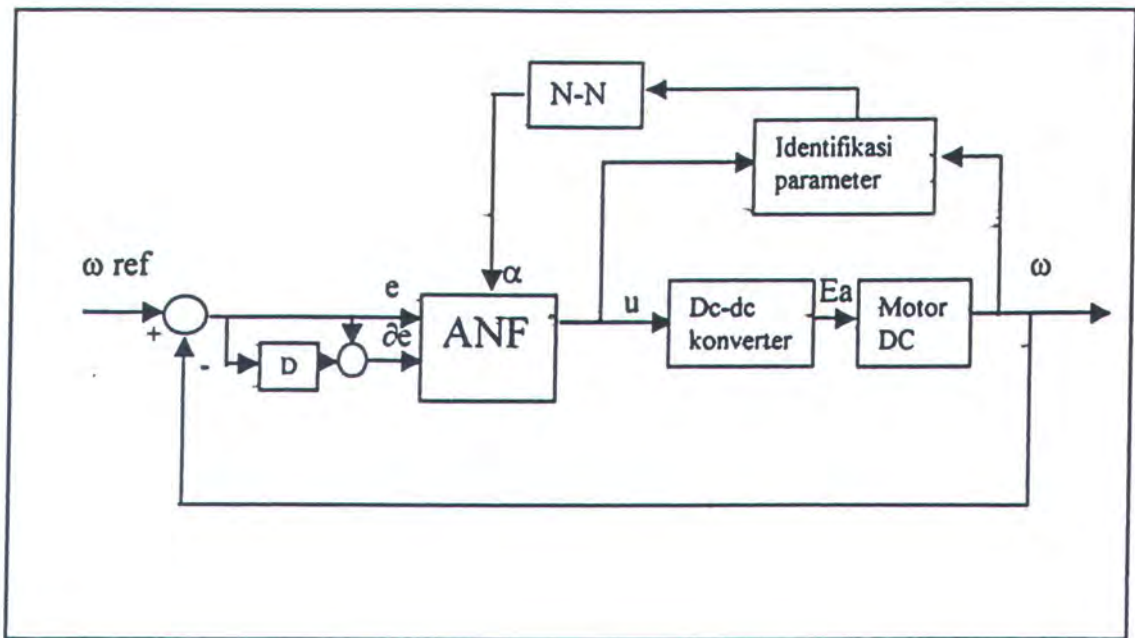


Gambar 3.2 Arsitektur jaringan syaraf tiruan untuk pembelajaran alfa

Nilai-nilai parameter dan nilai-nilai alfa (α) yang telah diperoleh selanjutnya dijadikan pembelajaran pada jaringan syaraf tiruan dimana i adalah jumlah data yang akan dijadikan proses pembelajaran dengan menggunakan cara supervised. Pembelajaran dilakukan sampai dicapai error relatif yang kecil. Nilai dari pembobot jaringan syaraf tiruan ketika telah mencapai error relatif yang kecil pada tiap perubahan beban dicatat dan ditampilkan dalam bentuk grafik. Sehingga nantinya akan didapatkan hubungan antara pembobot dengan perubahan beban. Sehingga jika terjadi perubahan beban maka dapat ditemukan nilai pembobot jaringan yang bersesuaian.

3.1.5. Simulasi sistem keseluruhan

Simulasi sistem keseluruhan dengan kontroler Fuzzy Adaptif yang telah memperoleh input selain error dan delta error berupa nilai alfa yang akan berubah-ubah secara otomatis sesuai dengan perubahan beban. Nilai alfa ini menjadi masukan bagi fungsi koridor sehingga fungsi keanggotaannya juga akan berubah-ubah sesuai dengan perubahan beban, Oleh karena itu aksi kontrol u harus mampu mengikuti perubahan beban yang diberikan. Diharapkan nantinya output dari ANF(Adaptive-Neuro-Fuzzy) secara otomatis akan menghasilkan aksi kontrol bagi plant untuk mengatur kecepatan motor sesuai dengan performansi yang diinginkan. Blok diagram sistem secara keseluruhan dapat digambarkan sebagai berikut :



Gambar 3.3 Blok diagram sistem pengaturan kecepatan motor dc dengan kontroler ANF

3.2 METODE PENGUJIAN

Pengujian akan dilakukan dengan membandingkan hasil respon pada saat menggunakan kontroler yang memakai fungsi koridor dengan alfa sama dengan satu atau dengan kata lain menggunakan fuzzy yang fungsi keanggotaannya bersifat statis dengan kontroler fuzzy adaptif yang fungsi keanggotaannya dapat berubah-ubah secara otomatis karena telah melalui proses pembelajaran. Akan dilihat bagaimana kinerja dari kontroler bila terjadi perubahan setpoint dan beban.

“Dan kami perintahkan kepada manusia agar berbuat baik kepada ibu-
bapaknya. Ibunya mengandungnya dalam keadaan lemah yang
bertambah-tambah...”.

(Luqman :14)

BARU
PENGUJIAN DAN ANALISIS

BAB IV

PENGUJIAN DAN ANALISIS DATA

Pada penelitian ini kami akan membandingkan metode dengan fuzzy biasa dan fuzzy Adaptif untuk perubahan setpoint/beban. Untuk fuzzy biasa fungsi keanggotaannya bersifat statis. Dan untuk fuzzy Adaptif fungsi keanggotaannya akan berubah secara otomatis karena telah melalui proses pembelajaran.

4.1 PENGUJIAN DENGAN KONTROLER FUZZY BIASA

Dengan menggunakan persamaan (2.60), (2.62), (2.64) dan (2.68) diperoleh persamaan differensial motor. Pada pengujian ini terlebih dahulu dilakukan inisialisasi untuk menyelesaikan persamaan differensial motor dengan metode runge kutta. Adapun persamaannya sebagai berikut :

$$\dot{i}_a = -\left(\frac{R_a}{L_a}\right)i_a - \frac{(K_b |\sin \omega_m t|)}{L_a} + \frac{E_a}{L_a} \quad (4.1)$$

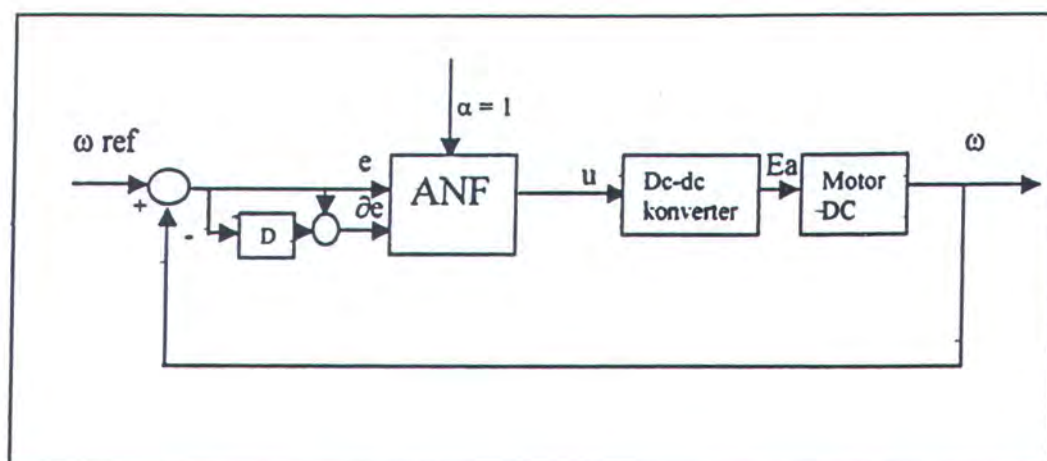
$$\dot{\omega}_m = \left(\frac{K_m}{J_m}\right)i_a - \left(\frac{B_m}{J_m}\right)\omega_m - \frac{T_L}{J_m} \quad (4.2)$$

Nilai parameter-parameter dari DC-DC Konverter dapat ditentukan sebagai berikut :

Periode DC-DC Konverter (T) = 0,01 detik

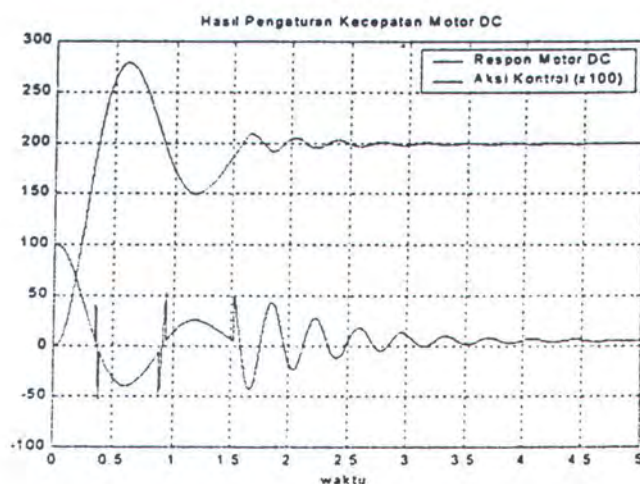
Tegangan referensi = 48 volt

Time sampling = 0,001 detik

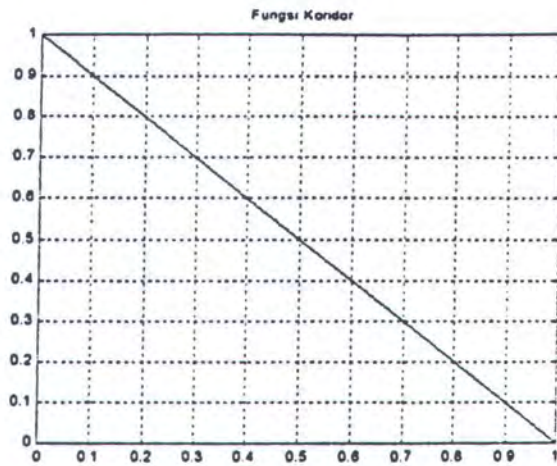


Gambar 4.1 Blok diagram sistem pengaturan kecepatan motor dc dengan kontroler fuzzy biasa

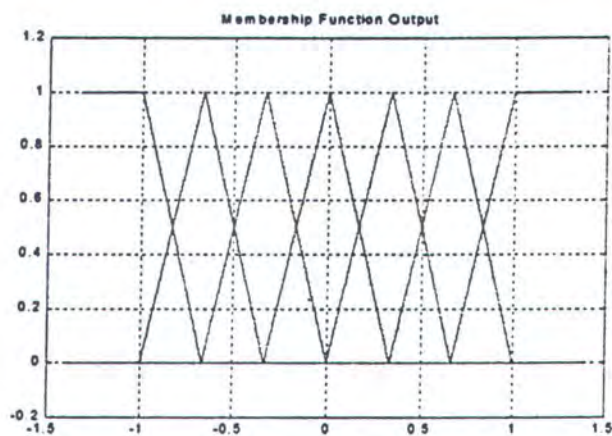
Untuk fuzzy biasa pengujian yang dilakukan bertujuan untuk melihat performansinya melalui respon yang dihasilkan jika fungsi keanggotaannya diubah-ubah. Dalam hal ini kami menggunakan fungsi koridor untuk mengatur fungsi keanggotaan. Karena masih menggunakan fuzzy biasa sehingga fungsi keanggotaannya masih uniform yaitu alfa sama dengan satu. Setelah melakukan simulasi program diperoleh hasil seperti pada ditunjukkan pada gambar dibawah ini



Gambar 4.2 Karakteristik respon dan aksi kontrol untuk alfa = 1



Gambar 4.3 Fungsi koridor untuk $\alpha = 1$

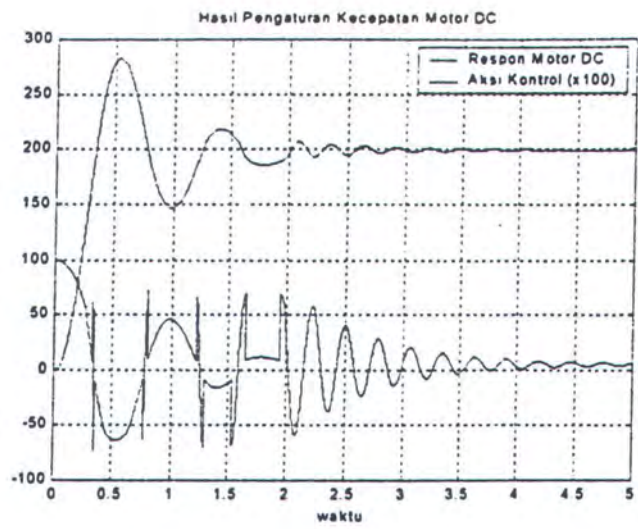


Gambar 4.4 Fungsi keanggotaan untuk $\alpha = 1$

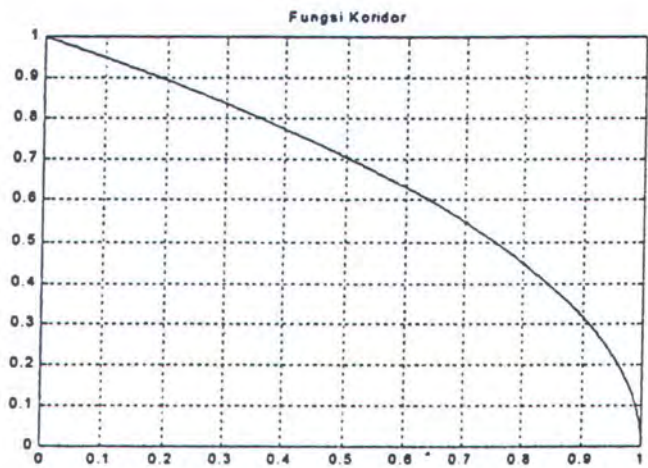
Kriteria performansinya sebagai berikut :

Settling Time	= 2.641 Detik
Rise Time	= 0.335 Detik
Mov	= 39.3647 %
Error	= 0.94441 rpm
Error Steady State	= 0.47221 %

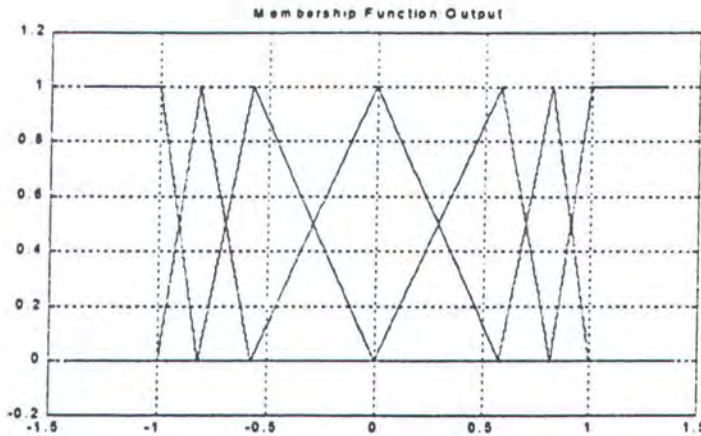
Jika fungsi keanggotaannya diubah secara manual yaitu α sama dengan 0,5 maka akan diperoleh hasil seperti gambar dibawah ini.



Gambar 4.5 Karakteristik respon dan aksi kontrol untuk $\alpha = 0.5$



Gambar 4.6 Fungsi koridor untuk $\alpha = 0.5$



Gambar 4.7 Fungsi Keanggotaan untuk $\alpha = 0.5$

Kriteria performansinya sebagai berikut :

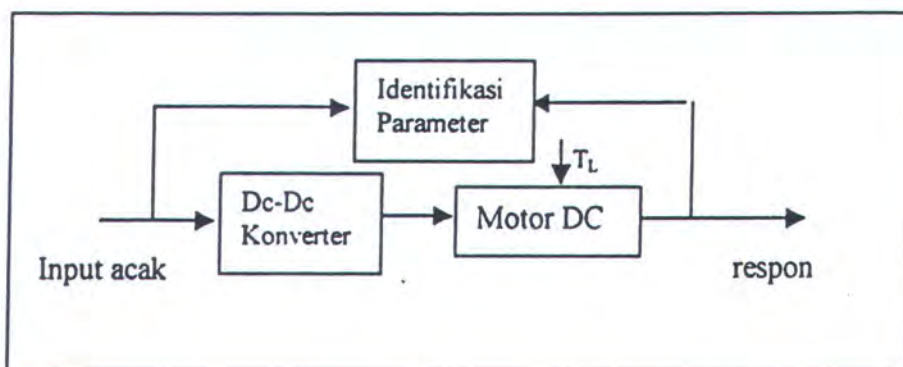
Settling Time	= 2.951 Detik
Rise Time	= 0.314 Detik
Mov	= 41.5512 %
Error	= 0.67339 rpm
Error Steady State	= 0.3367 %

Dari gambar diatas menunjukkan bahwa dengan perubahan alfa (fungsi keanggotaan) dapat memperbaiki performansi system, kecuali nilai overshootnya masih lebih tinggi.

4.2 PENGUJIAN DENGAN KONTROLER ADAPTIF NEURO-FUZZY

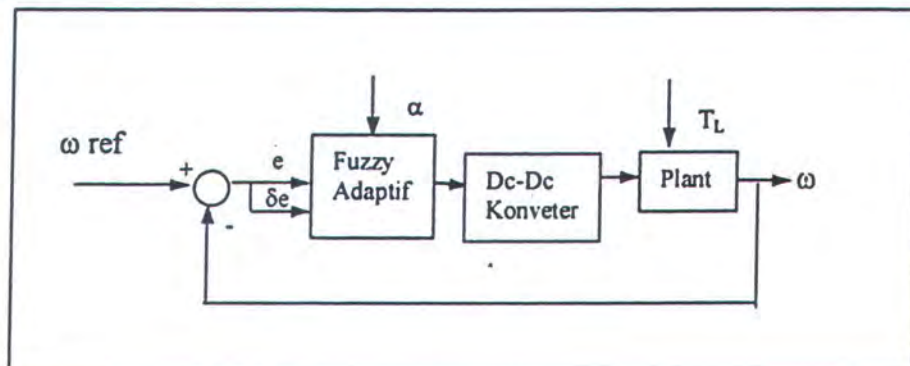
4.2.1. Proses Identifikasi Parameter dan Pembelajaran

Pada proses ini terlebih dahulu dilakukan identifikasi parameter terhadap plant kami yaitu dc-dc konverter dan motor dc. Dengan metode off-line pada system open loop kami melakukan proses identifikasi parameter dengan cara rekursif least square (RLS) untuk memperoleh beberapa nilai parameter-parameter a_1, a_2, b_1, b_2 untuk beberapa perubahan beban. Adapun blok diagram proses identifikasi seperti pada gambar dibawah ini.



Gambar 4.8 Blok diagram untuk identifikasi parameter

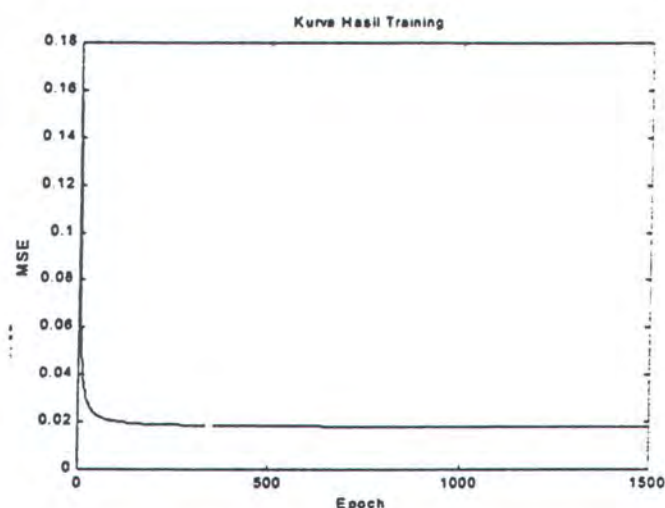
Selanjutnya kami mencari nilai alfa untuk beberapa kasus perubahan data yang sama sehingga parameter-parameter yang kami peroleh pada proses identifikasi parameter sebagai input dan nilai beberapa alfa sebagai output yang selanjutnya kami gunakan dalam pembelajaran. Adapun blok diagram untuk mencari nilai alfa seperti gambar dibawah ini.



Gambar 4.9 Blok diagram untuk mencari nilai alfa

Selanjutnya kami melakukan pembelajaran untuk memperoleh nilai pembobot yang bersesuaian sampai dicapai error yang relatif kecil. Hasil yang kami peroleh dapat dilihat pada gambar 4.7 yang memperlihatkan hasil training . Adapun data yang kami gunakan terdiri dari :

Data = 20 buah
 Input = 4
 Lapisan hidden = 2
 Output = 1
 Koefisien learning = 0,3
 Momentum = 0.1

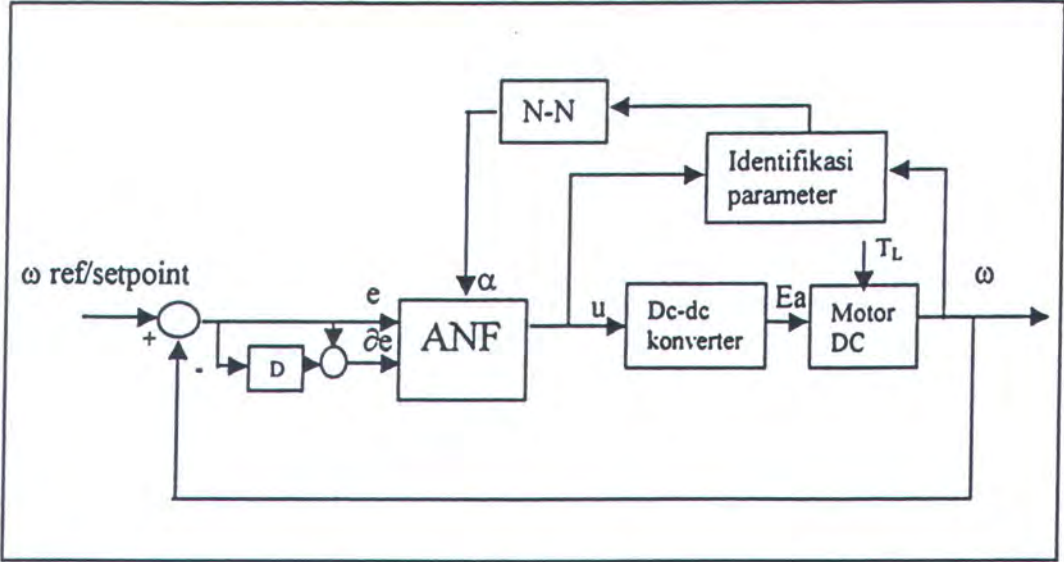


Gambar 4.10 Kurva training ANN mencari hubungan alfa dan beban

Dari hasil training diatas memperlihatkan nilai MSE yang semakin kecil

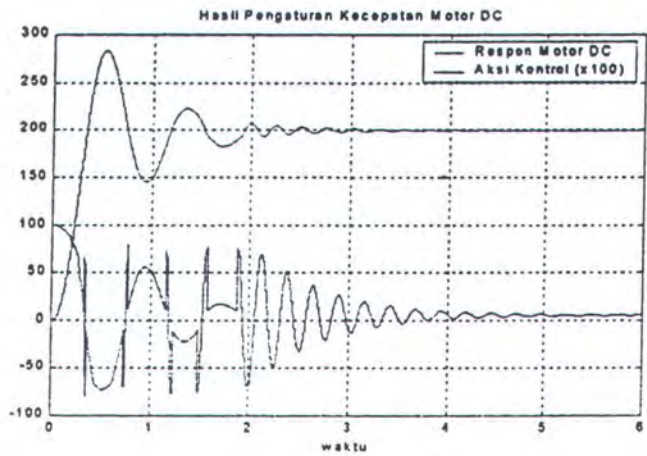
4.2.2. Pengujian dengan Adaptif Neuro-Fuzzy

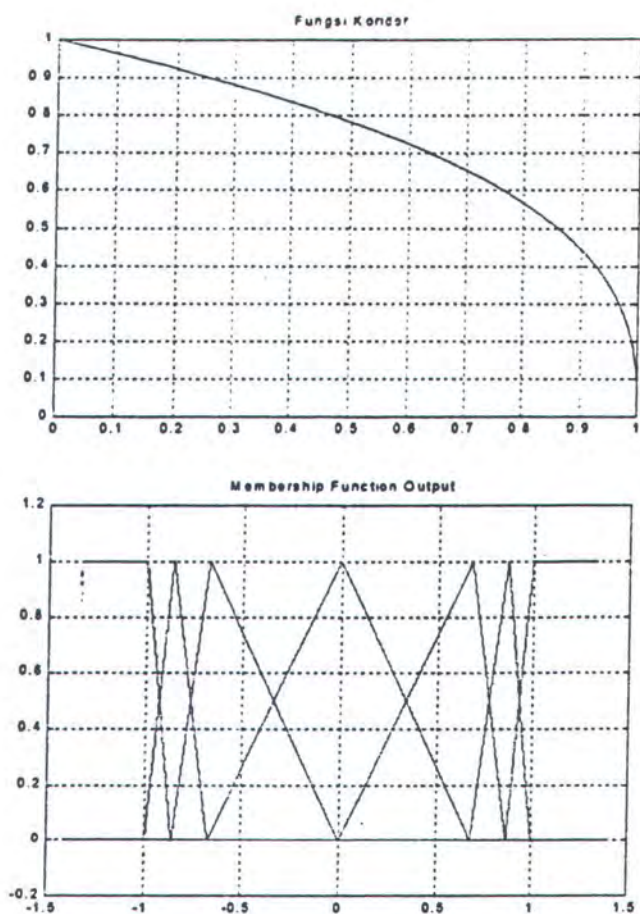
Pemilihan nilai alfa yang terbaik berdasarkan skala prioritas yang kami utamakan, secara berturut-turut yaitu Error Steady state, Rise Time, Settling Time dan yang terakhir adalah Maksimum overshoot. Sehingga nantinya secara otomatis alfa yang digunakan adalah yang terbaik berdasarkan urutan performansi yang kami inginkan. Blok diagram dari Kontroler ANF dapat ditunjukkan pada gambar dibawah ini.



Gambar 4.11 Blok diagram sistem pengaturan kecepatan motor dc dengan kontroler ANF

Hasil akhir kontoler Adaptif Neuro Fuzzy seperti ditunjukkan pada gambar 4.12.





Gambar 4.12 Karakteristik respon, aksi kontrol, fungsi koridor dan fungsi keanggotaan setelah pembelajaran

Kriteria Performansi seperti berikut :

Settling Time	= 2.913 Detik
Rise Time	= 0.31 Detik
Mov	= 41.916 %
Error	= 0.53358 rpm
Error Steady State	= 0.26679 %

Dengan hasil diatas menunjukkan bahwa dengan ANN maka secara otomatis kontroler ANF akan merubah nilai alfa ke harga yang paling mendekati dengan kriteria performansi yang kita inginkan.

Untuk melihat dengan jelas perbedaan performansi yang dihasilkan dapat di tabelkan sebagai berikut :

Tabel 4.1 Kriteria performansi untuk beberapa pengujian

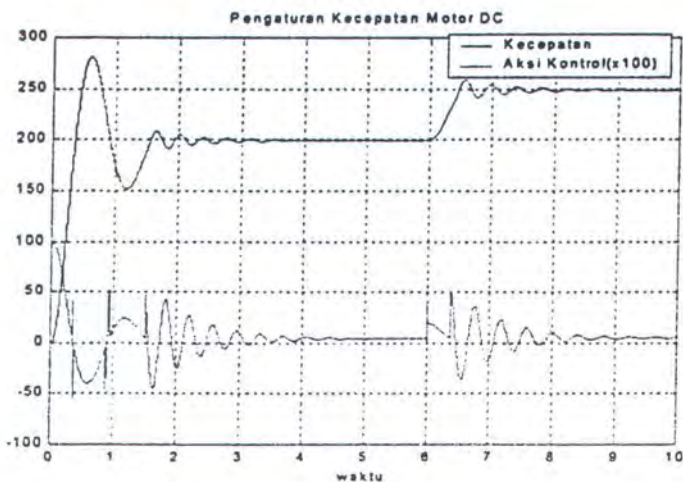
No	Jenis Pengujian Kontroler	Kriteria Performansi				
		% Ess	Error (rpm)	Tr (s)	Ts (s)	% Mp
1.	Fuzzy Biasa alfa = 1	0.472	0.944	0.335	2.641	39.365
2.	Fuzzy Biasa alfa = 0.5	0.337	0.673	0.314	2.951	41.551
3.	ANF	0.267	0.533	0.31	2.913	41.916

Dari table diatas dapat ditarik kesimpulan bahwa Kontroler ANF memiliki performansi yang terbaik ditinjau dari %Ess, Error, Rise Time walaupun untuk Settling Time yang terbaik ialah pada kontroler Fuzzy Biasa dengan alfa = 1.

4.3. PENGUJIAN UNTUK PERUBAHAN SETPOINT

A. Pada Kontroler Fuzzy Biasa

Pada pengujian ini pada T = 6 diadakan perubahan nilai setpoint dari 200 rpm menjadi 250 rpm. Hasil simulasinya dapat ditunjukkan pada gambar 4.13.

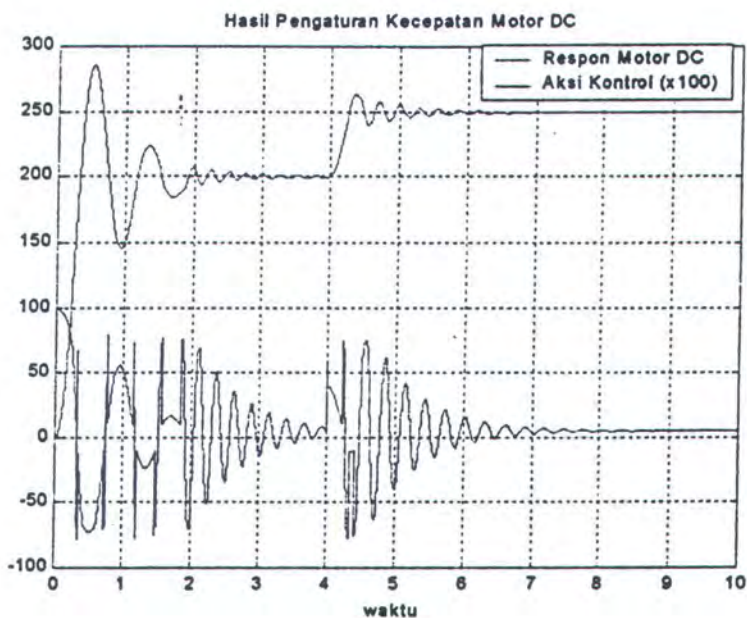


Gambar 4.13. Karakteristik respon dan aksi kontrol untuk perubahan setpoint

Kriteria Performansi seperti berikut :

Settling Time	= 7.628 Detik
Rise Time	= 0.328 Detik
Mov	= 12.4188 %
Error	= 1.3263 rpm
Error Steady State	= 0.53051 %
Kriteria IAE	= 58098.4046

B. Kontroler ANF



Gambar 4.14 Karakteristik respon dan aksi kontrol untuk perubahan setpoint setelah pembelajaran

Kriteria Performansi seperti berikut :

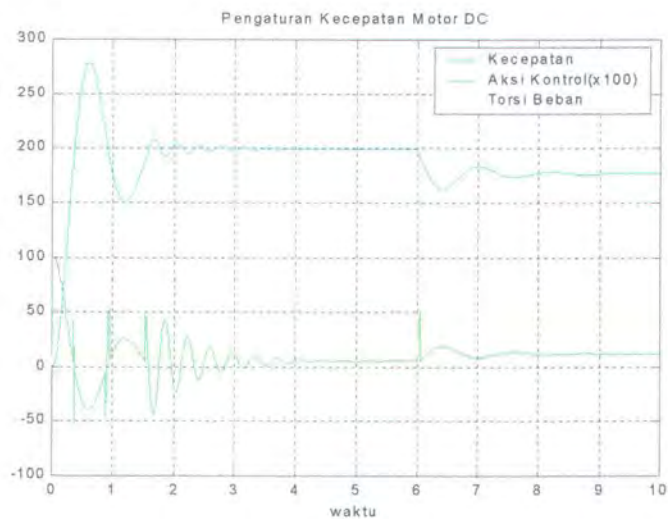
Settling Time	= 5.735 Detik
Rise Time	= 0.304 Detik
Mov	= 14.3957 %
Error	= 0.63682 rpm
Error Steady State	= 0.25473 %
Kriteria IAE	= 53570.5436

Dari gambar diatas dapat dilihat bahwa kontroler Fuzzy Biasa dan kontroler ANF mampu mengadaptasi keadaan jika terjadi perubahan setpoint. Namun secara keseluruhan kontroler ANF mempunyai performansi yang lebih baik dibanding kontroler Fuzzy Biasa, kecuali % Max Overshoot.

4.4. PENGUJIAN UNTUK PERUBAHAN BEBAN

A. Kontroler Fuzzy Biasa

Pada pengujian dengan perubahan beban yang tetap pada $T = 6$ dimana bebannya dinaikkan 30 % dari beban nominal. Hasil simulasinya dapat ditunjukkan pada gambar dibawah ini.

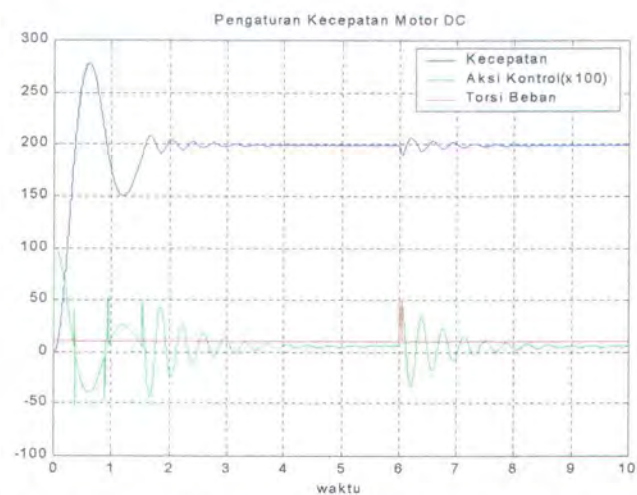


Gambar 4.15 Karakteristik respon dan aksi kontrol untuk perubahan beban yang tetap

Kriteria Performansi seperti berikut :

Settling Time	= 7.765Detik
Rise Time	= 0.335Detik
Mov	= 39.3647%
Error	= 23.6478rpm
Error Steady State	= 11.8239%
Kriteria IAE	= 98112.1017

Pada pengujian dengan perubahan beban yang sesaat pada $T = 6$ dimana bebannya dinaikkan 30 % dari beban nominal. Hasil simulasinya dapat ditunjukkan pada gambar dibawah ini.



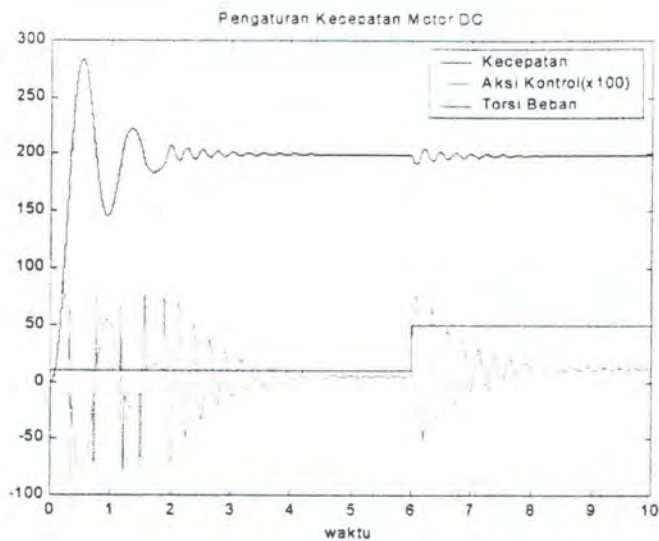
Gambar 4.16 Karakteristik respon dan aksi kontrol untuk perubahan beban sesaat

Kriteria performansinya sebagai berikut :

Settling Time	= 7.153 Detik
Rise Time	= 0.337 Detik
Mov	= 39.3647 %
Error	= 1.0293 rpm
Error Steady State	= 0.51466 %
Kriteria IAE	= 53554.8311

B. Kontroler ANF

Pada pengujian dengan perubahan beban yang tetap pada $T = 6$ dimana bebannya dinaikkan 30 % dari beban nominal. Hasil simulasinya dapat ditunjukkan pada gambar dibawah ini.

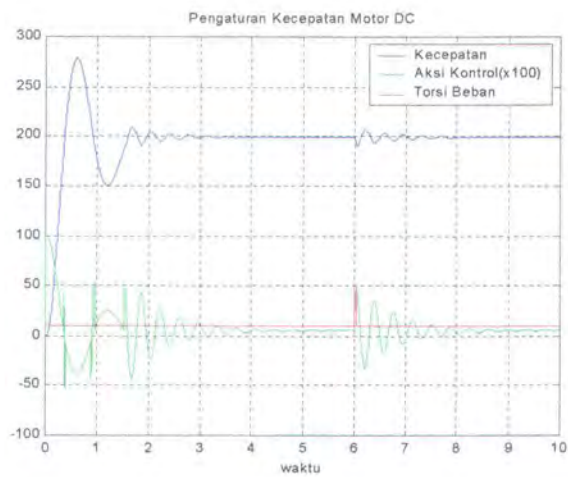


Gambar 4.17. Karakteristik respon dan aksi kontrol untuk perubahan beban yang tetap setelah pembelajaran

Kriteria performansinya sebagai berikut :

Settling Time	= 6.996 Detik
Rise Time	= 0.31 Detik
Mov	= 41.916 %
Error	= 1.2129 rpm
Error Steady State	= 0.60643 %
Kriteria IAE	= 50535.0331

Pada pengujian dengan perubahan beban yang sesaat pada $T = 6$ dimana bebannya dinaikkan 30 % dari beban nominal. Hasil simulasinya dapat ditunjukkan pada gambar dibawah ini.



Gambar 4.18 Karakteristik respon dan aksi kontrol untuk perubahan beban sesaat setelah pembelajaran

Kriteria performansinya sebagai berikut :

Settling Time	= 7.153 Detik
Rise Time	= 0.335 Detik
Mov	= 39.3647 %
Error	= 1.0292 rpm
Error Steady State	= 0.51462 %
Kriteria IAE	= 53554.0127

Dapat dilihat bahwa kontroler ANF mampu mengantisipasi keadaan lebih baik dibanding kontroler Fuzzy Biasa jika terjadi perubahan beban baik tetap maupun sesaat, kecuali pada % max Overshoot .

Untuk melihat dengan jelas perbedaan performansi yang dihasilkan dapat di tabelkan sebagai berikut :

Tabel 4.2 Kriteria performansi untuk pengujian Kontroler untuk perubahan setpoint dan beban

No	Jenis Pengujian	Kriteria Performansi berdasarkan prioritas				
		IAE	% Ess	Error	Tr(s)	Ts(s)
1	<u>Perubahan Setpoint</u> Kontroler Fuzzy Biasa	58098.404	0.5305	1.3263	0.328	7.628
	Kontroler ANF	53570.543	0.2547	0.6368	0.304	5.735
2	<u>Perubahan Beban tetap</u> Kontroler Fuzzy Biasa	98112.102	11.824	23.648	0.335	7.765
	Kontroler ANF	50535.033	0.6064	1.2129	0.31	6.996
3	<u>Perubahan Beban sesaat</u> Kontroler Fuzzy Biasa	53554.831	0.5147	1.0293	0.337	7.153
	Kontroler ANF	53554.012	0.5146	1.0292	0.335	7.153

Dari table diatas dapat disimpulkan bahwa secara keseluruhan untuk berbagai kondisi gangguan seperti diatas kontroler ANF memiliki performansi yang lebih baik dari kontroler Fuzzy Biasa ditinjau dari Kriteria IAE, %Ess, Error, Rise Time dan Settling Time.

“Ya Allah, ilhamkanlah kepadaku kecerdasan dan lindungilah aku dari
kejahatan nafsuku”
(Risalah Do’a)

BAB V

KESIMPULAN DAN SARAN

BAB V

KESIMPULAN DAN SARAN

5.1. KESIMPULAN

Berdasarkan pengujian dan pembahasan diatas dapat diambil beberapa kesimpulan sebagai berikut :

1. Dengan menggunakan fungsi koridor dalam mengubah range fungsi keanggotaan dapat memperbaiki performansi.
2. Secara otomatis kontroler ANF dapat digunakan sesuai kriteria performansi yang kita inginkan.
3. Kontroler Adaptif Neuro Fuzzy (ANF) menghasilkan performansi yang lebih baik dibanding dengan Kontroler Fuzzy Biasa berdasarkan IAE, % Ess, Rise Time (Tr), Settling Time (Ts).
4. Kontroler ANF dianggap robust terhadap perubahan setpoint dan perubahan beban karena dapat mempertahankan performansinya.

5.2. SARAN

1. Karena pada training data nilai MSE masih belum betul mendekati nol sehingga hasilnya untuk beberapa pengujian nilai alfa belum begitu berubah untuk berbagai perubahan setpoint/beban. Untuk penelitian selanjutnya mungkin dilakukan data yang lebih banyak dan bervariasi.
2. Karena penelitian ini terbatas pada skala simulasi, ada baiknya jika pada penelitian selanjutnya diimplementasikan dalam real plant.

dengan ilmu.....

hidup menjadi mudah

dengan agama.....

hidup menjadi terarah dan bermakna

DAFTAR PUSTAKA

DAFTAR PUSTAKA

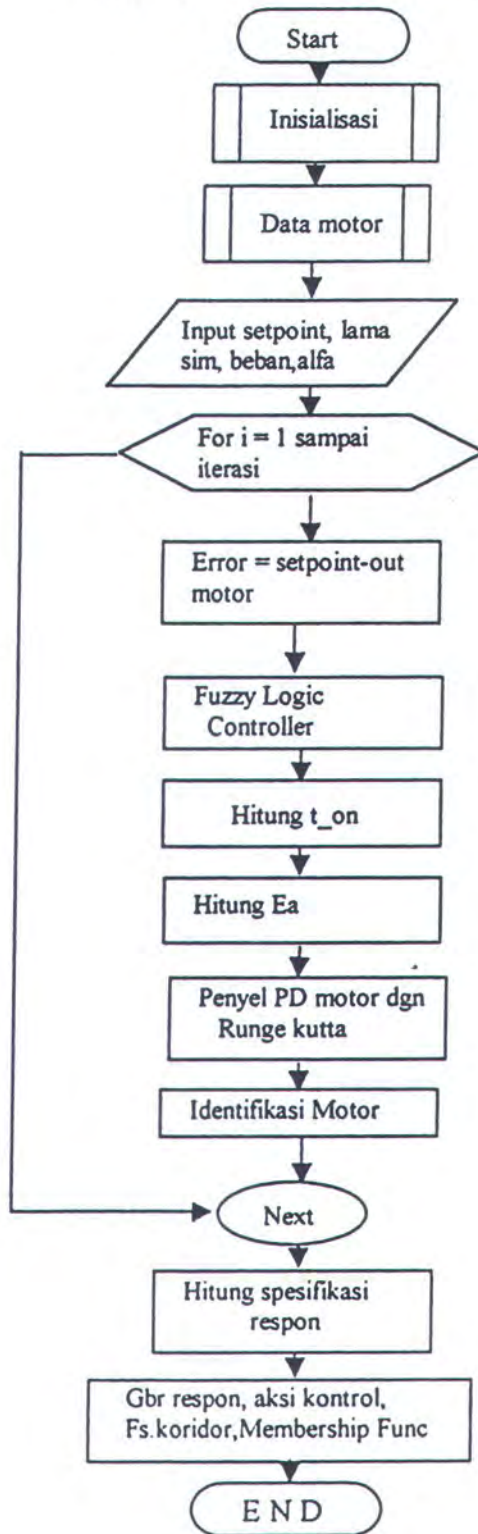
1. Dubey K, Gopal. [1989], "Power Electronics Controlled Drives", Prentice Hall.
2. Gupta, Tarup., et al. [1997], "Implementation of Fuzzy Controller for Dc-Dc Converters Using an Inexpensive 8-b Microcontroller", IEEE Transaction on Industrial Electronics, Vol 44, N0.5
3. Kuswadi, Son., Dkk. [1998], "Elektronika Terapan", Pusat Pengembangan Pendidikan Politeknik, Depdikbud, Bandung.
4. Patterson, Dan W. [1996], "Artificial Neural Network : Theory and Applications, Prentice Hall, Singapore
5. P. Mattavelli., et al. [1997], "General Purpose Fuzzy Controller for Dc-Dc Converter", IEEE Transaction on Power Electronics, Vol. 12, No. 1.
6. Rashid, Muhammad. "Power Electronics : Circuit, Devices, and Applications", Prentice Hall International Edition, 2-rd edition.
7. Subiyanta, Erfan. [2001], "Simulasi Pengaturan Kecepatan Pada Motor DC Menggunakan Indirect Adaptive Fuzzy Logic Controller dengan metode Inversi Kausal", Tugas Akhir, Fakultas Teknik Elektro, ITS, Surabaya.
8. Teng lin, Chin., Lee, George, C.S. [1996], "Neural Fuzzy Systems", Prentice Hall International,
9. Yan, Jun., Ryan, Michael., [1994] " Using Fuzzy Logiz", Prentice Hall International.



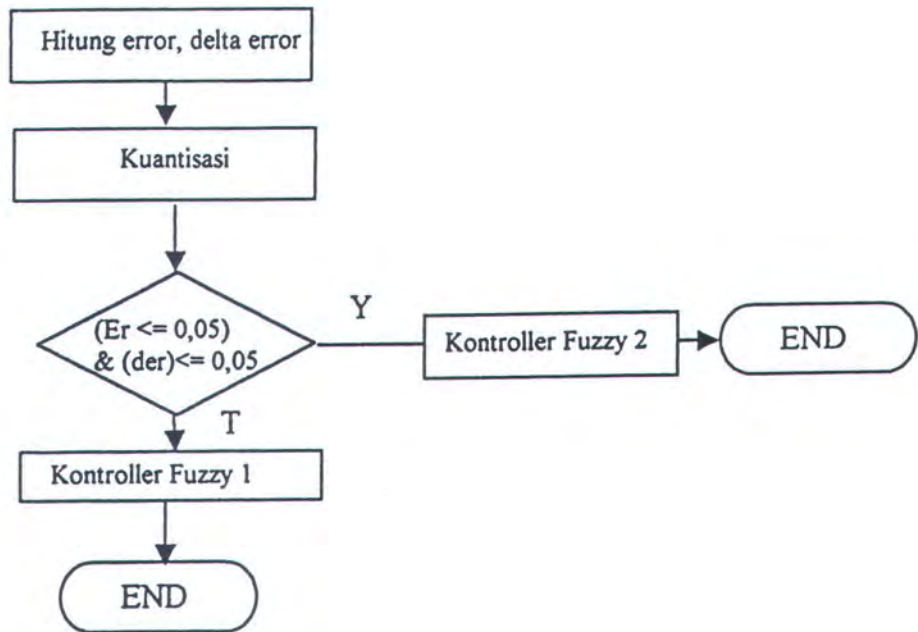
LAMPIRAN

Lampiran A

FLOWCHART "SIMULASI AWAL"



FUZZY CONTROLLER



Lampiran C

FLOWCHART "SIMULASI AKHIR"

